ATARI ST COMPUTER

Die Fachzeitschrift für den ATARI-ST Anwender.

Dezember 88

DM 7,-

Ös. 56,- Sfr. 7,-







CALAMUS

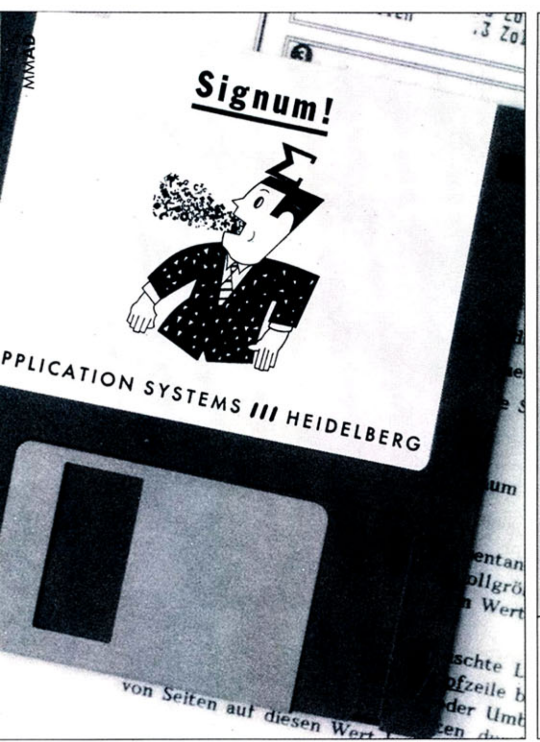
Professionelle DTP?

Schnelle 3D-Grafik



Turbo C contra Laser C

Die undokumentierten Befehle des OMIKRON.BASIC







Alt: Signum!Zwei

Natürlich ist *Signum!Zwei* nicht so alt, wie man aussieht, wenn man diffizile Texte bearbeiten möchte und es nicht zur Hand hat. Das Programm ist seit etwa zwei Jahren auf dem Markt und in seiner Version „Zwei" mit umfangreichen Attributen gesegnet, die zu jedermanns Textkomfort dienlich sind: zum Beispiel Mehrspaltensatz, freie Formeldefinition, automatische Silbentrennung und Bildintegration.

Über 350 verschiedene Zeichensätze sind mittlerweile für *Signum!* erhältlich und erlauben ein souveränes Gestalten von Manuskripttexten und Formeln, die bereits auf einem 9-Nadel-Drucker hervorragend aussehen. Um so schöner natürlich auf einem 24-Nadel- oder einem Laser-Drucker.

Signum! ist bei der Arbeit behilflich, das bestätigen Hunderte von Wissenschaftlern, Textern oder Autoren. Auf Anfrage senden wir Ihnen gerne umfangreiches Informationsmaterial, Probeausdrucke und einen Händlernachweis, damit Sie wissen, wo Sie *Signum!* bekommen können.

Lesen und lesen lassen.

Wie das so ist: Da sitzt man da und grübelt über den richtigen Software-Support. Man hat sich schon oft mit den Anwendern unterhalten, telefoniert, hunderte, ja tausende von Briefen geschrieben.

In punkto Signum! hat sich dabei in zwei Jahren eine unglaubliche Fülle von Informationen angesammelt, die den Umfang und den Inhalt jeden Handbuches sprengen würde.

Darum haben wir uns entschlossen, das definitive Buch zum Thema herauszubringen, denn wer könnte dafür kompetenter sein, als Volker Ritzhaupt, der Signum! von Anfang an betreut. Kurzum: *„Das Signum!-Buch"* ist da!

Im Format DIN A 5 auf 432 Seiten zeigt das Buch dem Anfänger alles vom ersten Schritt an, dazu praktische Beispiele und Bilder. Der Fortgeschrittene findet die kompletten Tips und Tricks und einen umfangreichen Nachschlageteil. *„Das Signum!-Buch"* ist stabil gebunden, fadengeheftet und kostet 59,– DM.

Natürlich gibt es noch ein zweites Buch, wir ahnten es schon, das enthält alle Signum!-Zeichensätze, die bisher verfügbar sind. Sie sind übersichtlich dargestellt und werden in ihrer Anwendung in der Gestaltung gezeigt. Dieses Buch ist auch von Volker Ritzhaupt und hat auch einen Namen: *„351 Signum!-Zeichensätze. Ein Buch zur Gestaltung".* Es ist im Großformat, Paperback, hat 232 Seiten und kostet 29,– DM.

Die Bücher sind im Computerfach- und im Buchhandel erhältlich oder direkt bei uns.

Neu: Wie Daisy ihre Post macht.

Daisy macht es natürlich wie alle anderen: sie klebt eine Marke auf den Umschlag und der Brief geht ab in den Kasten. Das kann ihr auch kein Computer abnehmen, geschweige denn eine Software.

Dennoch: Daisy hat mit *Daily Mail* eine Software in der Hand, mit der sie spielend einen Brief mit unterschiedlichen Adressen versehen kann. Die Adressen befinden sich in einer Datenbank.
Unter anderem.

Briefköpfe können mit automatischer Absenderangabe angegeben werden. Briefe schreibt sie genauso unkompliziert wie mit einer normalen Textverarbeitung. Serienbrief und Sammelausdruck sind möglich, darüber hinaus verfügt *Daily Mail* über eine Schnittstelle zur Datenbank ADIMENS.
Zum Beispiel.

Daily Mail ist, der Autor dieser Zeilen bestätigt es hiermit, herzerfrischend einfach und logisch aufgebaut, ein Programm ohne großes Hin und Her, ideal für den täglichen Gebrauch. Mehr darüber verrät übrigens unser Software-Info „Daily Mail", das wir auf Anfrage gerne zusenden.
Aber bitte.

Die Kunst der ST-Software



Englerstraße 3
D-6900 Heidelberg
Telefon (06221) 300002
Telefax (06221) 300389

Vertrieb in der Schweiz:
Computer Trend AG
Langstrasse 31
CH-8021 Zürich

EDITORIAL

Schneller...

... schnell hoch in die nächste Abteilung. Hä,..uff,...puff, puff. Mein Puls spielt verrückt. Schnell... Ich merke schon, ich bin nicht mehr der sportliche und schlanke Mann von damals. (Ja, damals hätte mir das überhaupt nichts ausgemacht. Da hätte ich das auf die Schnelle erledigt!!! Ja, diese alte, schöne Zeit!) Was fehlt noch? Einen schnellen Blick auf die Liste und weiter. Schneller... Verdammt, ich komm zu spät. Schnell reagieren. Konzentration ... schnell. Schnelles Denken. Ja, stimmt, um die Ecke, da war's. Ja sicher, da könnten sie es wahrscheinlich auch haben. Schnell rüber. Schnell fragen, bevor irgendjemand anderes kommt und es mir wieder vor der Nase wegschnappt! Schnell, gnädige Frau, suchen Sie! Schnelle Antwort! Schneller Frust! Schnell laufe ich weiter. Schnell geht es mir durch den Kopf, wie schnell ich mich entschieden habe, all die Dinge auszusuchen. Ich dachte, es ging alles viel schneller. Große Enttäuschung! Nichts geht schnell, wenn man sich auf die Schnelle in irgendetwas hineinstürzt. Wieder schnell auf die Liste schauen. Was habe ich bis jetzt so gekauft. Ja, das hätten wir. Das war nicht so gut. Das war mit Sicherheit ein zu schneller Kauf. Mach weiter, schneller lesen. Das..., das auch.. schnell...! Aha, es fehlt immer noch das eine. Oh, und auch das andere. Der Puls steigt immer schneller. Mist, man merkt nicht, wie schnell die Zeit vergeht. Das ganze Jahr ist so schnell vergangen und jetzt läuft mir die Zeit davon. Schneller Blick auf die Uhr. Gleich fängt die Sportschau an. Ich muß schnell nach Hause. Schnell, bevor es die Nächsten kaufen. Schnell werden auch die Kassen zugemacht. Schnell will auch die Verkäuferin nach Hause gehen. Ein schneller Blick ins Portemonnaie. Schnell festgestellt, daß ziemlich schnell das Geld ausgegangen ist. Schnell durchkalkuliert, was noch zu besorgen ist. Es könnte reichen. Schnell wieder ins Kaufhaus. Schnell......, schnell...., schnell. Schnell wieder auf die Straße. Schnell weitergefahren. Schnell die Tür aufgemacht. Schnell gegrüßt. Schnell den Fernseher angemacht. Schnell hingesetzt. Schnell überlege ich mir, daß ich nächste Weihnacht nicht so schnell wieder auf die Schnelle einkaufen gehen werde.

Angenehme RUHIGE Weihnacht!

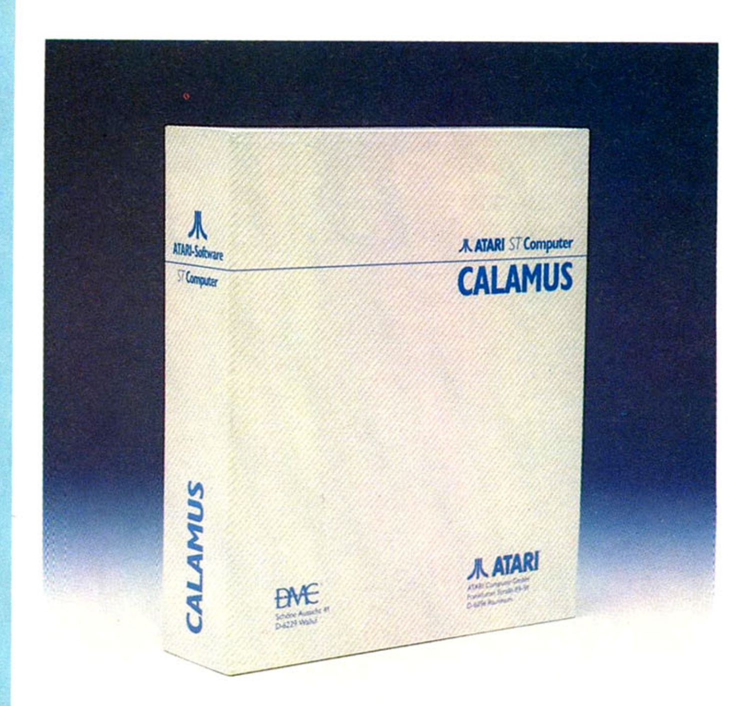
Marcelo Merino

I N H A L T

SOFTWARE

HARDWARE

GRUNDLAGEN



Calamus - professionelle DTP?

Seit über einem Jahr angekündigt und auf jeder wichtigen Messe als Vorabversion zu sehen, von vielen Leuten mit Vorschußlorbeeren versehen, mit ATARIs Roadshow auf Deutschlands Straßen, so sehen einige wahllos herausgegriffene Stichpunkte zum Thema Calamus aus. Jetzt sind die ersten Exemplare ausgeliefert worden. Grund genug, uns mit einem der wohl mächtigsten Softwarepakete für den ATARI ST zu beschäftigen.

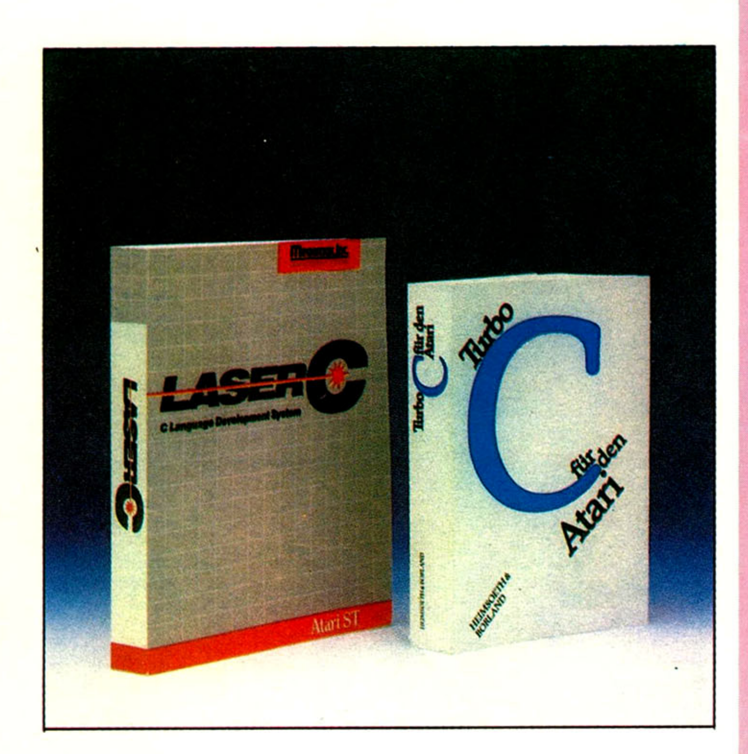
Seite **66**



ST Computer Weihnachtsbasar

Es weihnachtet sehr! Für Ihren diesjährigen Wunschzettel wollen wir Ihnen einige Geschenkideen präsentieren. Dabei werden wir quasi einen Rundumschlag im Bereich Computer führen und Ihnen Bücher, Software und Hardware präsentieren.

Seite **8**



Turbo C contra Laser C - Ring frei zur zweiten Runde

Bis vor kurzem war der Megamax C-Compiler der auf dem ATARI ST weitverbreitetste C-Compiler. Seinen Nachfolger Laser C haben wir bereits kurz vorgestellt. Ebenfalls kurz berichtet haben wir schon über seinen großen Konkurrenten, Turbo C, das ja bereits in der MS-DOS-Welt einen großen Namen hat. Wir werden beide Entwicklungssysteme miteinander vergleichen, um Ihnen Vor- und Nachteile aufzuzeigen.

Seite **50**

Keep Hacking Fremdtastaturen am ST

Sollten Sie die Werbung in dieser Zeitschrift ausführlich durchlesen, werden Sie feststellen, daß außer einer Originalplatine (und die gibt es auch schon z.B. im 19"-Gehäuse) eigentlich alle Peripheriegeräte von Fremdanbietern geliefert werden. Diese Produkte, Monitore, Laufwerke, Tastaturen usw. sind, um konkurrenzfähig zu sein, entweder viel billiger oder ganz einfach besser. Um den zwischenzeitlich großen Markt an Fremdtastaturen etwas zu durchleuchten, wollen wir Ihnen unsere Erfahrung mit vier verschiedenen Lösungen vermitteln, die alle ihre Vor- und Nachteile haben.

Seite **23**

PROJEKT

PROGRAMMIERPRAXIS

ANWENDUNGEN

AKTUELLES

RUBRIKEN

NEWS

Ein neuer Butler

James II

Dieses völlig überarbeitete Börsenspekulationsprogramm dürfte das Herz eines jeden Börsenspekulanten höher schlagen lassen, denn die IFA Köln berücksichtigte bei der Erstellung von JAMES II die Kritik, Wünsche und Ideen der Benutzer von JAMES, dem Vorgänger. JAMES II ist jetzt gänzlich in GEM eingebunden. Zusätzlich wurde ein 32-monatiges OVERBOUGHT/ OVERSOLD eingebaut, und POINT & FIGURE wurde um die Möglichkeiten des Korrekturfaktors, des frei wählbaren Zeitraums bzw. frei wählbaren Umkehrpunkts bereichert. Weiterhin kann dieses Programm einen RSI-Chart vorweisen wie auch 38 Tage-, 100 Tage- und 200 Tage-Oszillatoren. Da es für jeden Spekulanten von großem Interesse ist, den Kursverlauf einzelner Wirtschaftsunternehmen miteinander zu vergleichen, erhält er nun die Möglichkeit, Grafiken zweier Unternehmen übereinanderzulegen, um somit einen produktiven Vergleich ziehen zu können. Auf bearbeitungstechnischer Ebene wird dem Anwender die Möglichkeit geboten, Linien und Bögen zu zeichnen, Kommentare im Chart zu schreiben und Bilder abzuspeichern. Die Aktien lassen sich alphabetisch sortieren, und durch einen Korrekturmodus können erkannte Fehler berichtigt werden. James II kostet 298,- DM, wobei den Anwendern die Datenbank der IFA Köln, die jeden Tag mit den neuesten Daten gefüttert wird, rund um die Uhr zur Verfügung steht.

IFA Köln
Gutenbergstr. 73
5000 Köln 30
Tel.: 0221/520428

MIDI-SCAN

Mit ihrem Programm MIDI-SCAN feiert die Firma Soundlab ihr erstes Softwaredebut. SoundLab, die, im März 1987 gegründet, sich schon einen recht guten Namen mit professionellen Sounds für diverse Synthesizer (DX-/II, D50, TX-81Z, FB-01, Matrix-6, ESQ-1) gemacht haben, haben mit MIDI-SCAN einen MIDI-Monitor geschaffen, mit dem es möglich ist, die Daten, die über die Schnittstelle kommen, anzuzeigen und zu interpretieren. Mit diesem MIDI-Monitor/Dump-Programm können Sie sich alle Daten online im Original und im Klartext darstellen lassen, wobei die Events auch aufgezeichnet werden können. Außerdem bietet das Programm das Abspeichern, Laden und Abspielen von Midi-Events, Edieren, Speichern, Laden und Senden von Dump-Requests sowie ein Active-Dump-Utility. Zu dem Programm wird eine 44 seitige Lektüre mitgeliefert, die neben der MIDI-SCAN-Anleitung eine ausführliche und einfach zu verstehende Einführung in die Welt der MIDI-Events bietet.

SoundLab
Siechenangerstr. 7
8640 Kronach
Tel.: 09261/95508

Wohin mit dem alten Drucker?



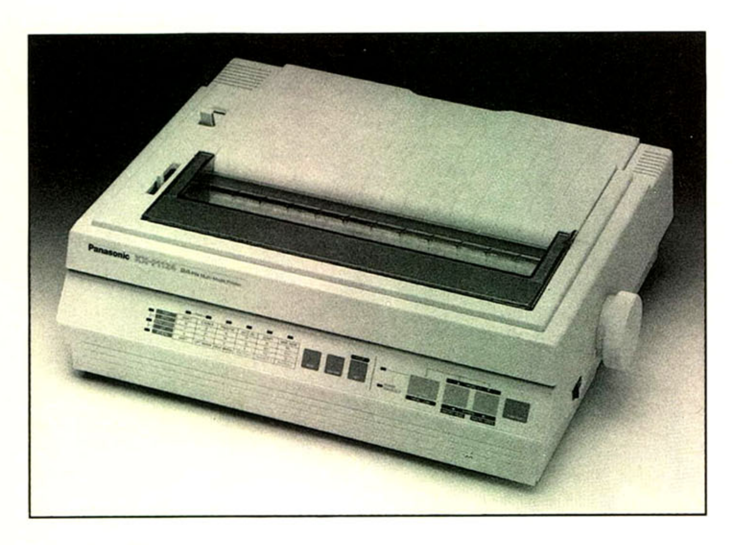
Wie Marktforscher berichteten, wird der Drucker-Austauschmarkt in den nächsten Jahren zunehmend an Bedeutung gewinnen. Schon jetzt macht sich der Trend bemerkbar.

Damit Anwender, die Drucker früherer Generation besitzen, einen solchen Austausch leichter und kosteneffektiver nutzen können, bietet ein Kreis von OKI-Fachhändlern seinen Kunden die Möglichkeit, alte, funktionsfähige Matrix- oder Typenraddrucker in Zahlung zu geben (gängige Marktkonditionen werden geboten). Hiermit wird vor allen Dingen eine Antwort auf die so oft gestellte Frage "Wohin mit dem alten Drucker?" gegeben. Im Austauschangebot stehen zwei OKI 9-Nadeldrucker, der MICROLINE 320 und der MICROLINE 321, zur Verfügung. Des weiteren wurde als besonderes Bonbon noch ein 24-Nadeldrucker oben draufgesetzt - der MICROLINE 391, ein Breitwagendrucker. Diese drei Drucker bieten ausgereiften Bedienungskomfort, flexible Papierhandhabung und hohe Druckgeschwindigkeiten.

OKIDATA GmbH
Hansaallee 187
4000 Düsseldorf 11
Tel.: 0211/59794-0

Neuer Drucker von Panasonic

Auf der Orgatechnik '88 hat Panasonic mit dem KX-P1124 einen neuen 24-Nadeldrucker



in angenehmem Design und kompakter Bauweise vorgestellt. Der KP-P1124 paßt aufgrund seiner Abmessungen auf jeden Arbeitsplatz und ist mit seinen 7 festinstallierten Schriftarten für die vielseitige Gestaltung von Schriftstücken auch in Korrespondenzqualität geeignet. Mit 24 Nadeln und einer superfeinen Auflösung von 360x360 dpi (Punkte pro Inch) können auch hochauflösende Grafiken bestens wiedergegeben werden. Der 11"-Matrixdrucker ist sogar in der Lage, DIN A4-Querformat zu verarbeiten. Zwei Druckeremulationen - EPSON LQ-2500 und IBM-Proprinter X24 - sowie die als Centronics-Verbindung ausgeführte Parallelschnittstelle und die als Option vefügbare serielle RS-232-Schnittstelle stehen für den weiten Einsatzbereich des neuen Druckers. Der Papiertransport erfolgt über einen umschaltbaren Zug-/Schubtraktor für Endlospapier - die Möglichkeit der einfachen Umstellung von Endlos- auf Einzelblattpapier erleichtert die Bedienung. Im Einzelblattbetrieb bleibt das Endlospapier im Printer, wobei in einem Durchgang ein Original und bis zu drei Blatt selbstdurchschreibendes Papier bedruckt werden können. Den Drucker gibt es schon seit November im Fachhandel.

Panasonic Deutschland GmbH
Winsbergring 15
2000 Hamburg 54
Tel.: 040/8549-0

Portabler Datenpuffer & Druckerumschalter 4 zu 1

Diese kleinen kompakten Datenpuffer sind nicht größer als eine Zigarettenschachtel. Die Stromversorgung erfolgt über eingebaute NiCd-Akkumulatoren, ein Lade-Adapter wird mitgeliefert. Sie können jede Art von Daten speichern: Text, Grafik, Programme...Die Aufnahme kann manuell in Form erfolgen, so daß gesendete Daten entweder den bisherigen Inhalt überschreiben oder hinten angefügt werden. Alternativ kann der Puffer mit Steuercodes wie ein RAM angesteuert werden. Die Wiedergabe erfolgt entweder zyklisch, einmalig den gesamten Pufferinhalts oder mit Hilfe von Steuercodes wahlfrei.

Die Übertragung der Daten geschieht über die bekannte RS-232-Schnittstelle, wobei die Baudrate wählbar zwischen 150 und 19200 Baud liegen kann. Dabei werden 7/8 Bits, 1/1.5/2 Stopbits, odd/even/no-Parity sowie die Handshakes XON/XOFF und DTR unterstützt. Zwischen Aufnahme und Wiedergabe können alle RS-232-Einstellungen geändert werden, so daß der Puffer auch als Formatkonverter einsetzbar ist. Als weiteres neues Produkt der Firma Wiesemann und Theis GmbH stellt sich ein Druckerumschalter vor, der keine Versorgungsspannung benötigt und die Umschaltung vollautomatisch erledigt. Möglich wurde dies durch den Einsatz eines kundenspezifischen ICs (Gate Array), das bei Siemens exklusiv gefertigt wird. Rund 1000 CMOS-Gatter sind so geschickt verdrahtet worden, daß die Versorgungsspannung automatisch aus den anliegenden Signalen bezogen wird. Die Anwendung wird damit genauso einfach wie bei einer Mehrfachsteckdose für Netzstecker: Anstecken und vergessen. Die Schnittstelle kostet 398,- DM und ist, wie auch der portable Puffer und eine kostenlose Broschüre *"Computersystem richtig installieren"*, bei folgender Adresse zu beziehen:

Wiesemann & Theis GmbH
Winchenbachstr. 3-5
5600 Wuppertal 2
Tel.: 0202/505077

SPC-Modula 1.4
multitaskingfähig

Seit der Düsseldorfer ATARI-Messe gibt es von advanced applications Vizcena GmbH eine neue Version ihres bekannten MODULA-2. Verbessert wurden folgende Eigenschaften: SPC-MODULA-2 ist voll multitaskingfähig. Es erlaubt ohne Klimmzüge volles Multitasking mit Fenster und Maus, so daß jedes Programm sofort in die Multitask-Umgebung integrierbar ist. Passend dazu wurde eine neue grafikgestützte Shell mit Icons entwickelt, mit der mehrere Anwendungen wie Editor, Compiler und so weiter gleichzeitig zur Verfügung stehen. SPC-Modula funktioniert völlig ohne Probleme mit den Großbildschirmen und auch die PAK68 macht ihm keine Schwierigkeiten, wodurch eine Geschwindigkeitssteigerung um den Faktor 5 zu erwarten ist. Die File-Utiliy wurde optimiert, ein Batch-Programm erlaubt komfortable Erstellung von Anweisungsfolgen, Die Windowschnittstelle SSWiS wurde erweitert und vieles mehr. Die neue Version 1.4 des SPC MODULA-2 gibt es bei

advanced applications
Viczena GmbH
Sperlingweg 19
7500 Karlsruhe 31
Tel.: 0721/700912

MALHEUR!

Seit etwa einem Monat wird das Buch SCHEIBENKLEISTER mit der Version 1.3 der KLEISTERSCHEIBE ausgeliefert; dabei hat sich auf der Masterdiskette ein kleiner Fehler eingeschlichen, der zwar leicht zu beheben, nichtsdestotrotz höchst ärgerlich ist. Das Problem: SED und TED liefen plötzlich nicht mehr auf Farbmonitor. Ursache: Ein einziges falsches Byte in den EXE-Dateien. Das nachfolgende kleine GFA BASIC-Programm behebt den Fehler. Die Autoren möchten sich auf diesem Weg bei allen Lesern entschuldigen.

Wer möchte, kann im SED die Zeile

```
Deftext 1,0,0,13
```

ersatzlos streichen (das ist die zweite Zeile im eigentlichen Programm, kurz nach den Kommentaren); sie kann in niedriger Auflösung zu unnötigen Konfusionen führen.

```
' Patcht SED.EXE und TED.EXE auf der KLEISTERSCHEIBE 1.3
'
Alert 1," Bitte KLEISTERSCHEIBE I in A: einlegen! | |",1," OK ",Erg%
Open "I",#1,"a:\liesmich.1_3"
Close #1                    ! Damit's auch die richtige Version ist
@Patch("a:\kap04.fld\sed.fld\sed.exe")
@Patch("a:\kap09.fld\ted.fld\ted.exe")
Print "SED.EXE und TED.EXE sind jetzt gepatcht."
Print
Print "Mit einer neuerlichen Entschuldigung -"
Print "keep hacking,"
Print Tab(30);"CB+AS"
'
Procedure Patch(Fi$)
  Open "U",#1,Fi$
  Seek #1,2041
  Out #1,8
  Close #1
Return
```

GFA BASIC komplett

Interessante Neuigkeiten sind aus dem Hause GFA zu melden. Der Preis für das GFA BASIC-Entwicklungssystem 2.0 (Interpreter & Compiler) beträgt neuerdings inklusive Dokumentation DM 49,90 und der Preis für das Entwicklungssystem 3.0 DM 198,-. Da der Compiler 3.0 noch nicht erhältlich ist (Erscheinungstermin Ende 1. Quartal '89), wird er gegen eine Bearbeitungsgebühr von DM 10,- nachgeliefert. Die Nachlieferung gilt übrigens auch für diejenigen, die schon im Besitz eines 3.0-Interpreters sind. Die einzige Bedingung ist, daß man bei GFA registriert ist. Aus diesem Grunde werden alle registrierten Kunden von GFA angeschrieben werden.

Ferner ist ein Upgrade vom Entwicklungssystem 2.0 auf Version 3.0 für DM 160,- möglich.

Zu jedem GFA-Produkt, das nicht bootfähig ist, wird ein Bootsektor-Virus-Checker mitgeliefert (z.B. GFA BASIC, GFA Assembler, GFA Draft+ usw.).

Für den wissentschaftlichen Bereich bietet GFA etwa ab Januar/Februar ein Statistik-Programm an.

GFA Systemtechnik GmbH
Heerdter Sandberg 30-32
4000 Düsseldorf
Tel.: 0211/55040

MSP190

Der ATARI-Portable



Basierend auf dem erfolgreichen 190ST hat IBP ein tragbares, ATARI-kompatibles Rechnersystem entwickelt, das in robuster 19"-Technik aufgebaut ist und damit leicht an kundenspezifische Anforderungen angepaßt werden kann. Das Gehäuse ist mit den Maßen 470x195x390 mm recht kompakt und mit dem Tragegriff leicht transportierbar.

Die Tastatur an der Vorderseite des Gerätes ist herausklappbar und optional auch als Folientastatur erhältlich. Der monochrome Bildschirm ist wahlweise für niedrige oder hohe Auflösung von 640x400 Punkten lieferbar, wobei auf Wunsch auch ein Farbmonitor eingebaut werden kann. Neben dem Monitor sind zwei 3.5"-Laufwerke mit je 720kB eingebaut. Im System ist ein Motherboard für wahlweise EUROBUS, ECB-, VME- oder SMP-Bus installiert. Das Board hat einen Steckplatz für die Rechnerkassette(190ST oder 190STE), einen für eine Harddisk 190HD sowie 5 Steckplätze mit je 4TE für Erweiterungskarten wie zum Beispiel AD/DA/IO. Die Spannungsversorgung der Einheit erfolgt über ein hochwertiges Netzteil und versorgt den Rechner und gegebenenfalls gesteckte Interfacekarten. Der Monitor wird in der hochauflösenden Ausführung mit 220V betrieben, während alle anderen Ausführungen mit 12V/DC aus dem Netzteil versorgt werden. Auf Anfrage kann das System für niedrige Eingangsgleichspannung ausgelegt werden. Zur Wärmeabfuhr ist ein geräuscharmer Lüfter eingebaut. Die Rechnerkassette sowie die Erweiterungskarten werden von hinten ins System eingesetzt. Dementsprechend sind die Peripherieanschlüsse der Rechnerkassette sowie eventuell installierte Interfacekarten auch von der Rückseite zugänglich.

IBP Gerätebau
Lilienthalstr. 13
3000 Hannover 1
Tel.: 0511/630963

ALLE JAHRE

ST Computer

Kommen Sie herein! Ja, Sie meine ich, den Herrn mit der roten Krawatte. Oder Sie, mit der Ledertasche unter dem Arm und der typischen Studentenhaltung. Kommen Sie einfach herein. Es ist Zeit für Bing Crosby und sein Jingle Bells. Weihnachten ist da! Es Zeit für den Vater oder die Omi oder die Freundin. Keiner denkt an etwas anderes als an's Schenken (vielleicht auch an's Beschenktwerden). Wir wollen hier jedoch keinen kritisch-soziologischen Diskurs über den Sinn oder Unsinn solcher Feierlichkeiten abhalten, wir möchten - weil Bing Crosby-Zeit ist - in unseren Basar einladen.



Nach fast drei Jahren Existenz der ATARI ST-Serie ist das Angebot an Produkten für diesen Rechner enorm groß geworden. Berge von Software verstecken mittlerweile die Hügel. Bis jetzt wurde soviel darüber geschrieben, daß die Bibliothek von Alexandria nicht alles aufnehmen könnte. Erweiterungen und Zusatzgeräte gibt es in jeder Form und Farbe, für jeden Geschmack ist etwas dabei. Die Frage, ob alles, was für die ST-Serie gemacht wurde, tauglich bzw. notwendig ist, können wir nur verneinen. Deshalb haben wir in all diesen Jahren eine Auswahl getroffen, die besten Produkte für den ATARI ST wurden kritisch getestet. Nun aber hinein in den Basar.

Erste Etage: Software

Damit sich ein Rechner durchsetzt, ist es das Wichtigste, daß genügend Software zur Verfügung steht. Ein Computer ist nach der Herstellung leblos, erst durch die Software beginnt er zu leben. Erinnern Sie sich noch an damals, als es einen CPM-Emulator für den ATARI ST gab? Man mußte sich mit dem alten *'Wordstar'* quälen. Aber es war aber der Anfang, und der Emulator war gar nicht so schlecht. Heute ist er immer noch zu bekommen, nämlich auf PD (in unserer PD-Sammlung die Nummer 5 und auf der PD-Diskette Nummer 6 gibt's ein paar Utilities für diesen Emulator).

Emulatoren Es, geht weiter...

Bleiben wir bei Emulatoren. Später, viel später, als die Softwareflut schon da war, kamen andere Emulatoren dazu. Die Notwendigkeit bestand zwar nicht unbedingt, aber der Bedarf irgendwie schon.

Aladin ist nicht nur der Name des Besitzers einer Wunderlampe, sondern auch ein Macintosh-Emulator der Spitzenklasse. Nichts hat eine gewisse Firma mit einem sehr fruchtigen Namen so geschmerzt wie die Existenz dieses Emulators. Unter *Aladin* laufen unzählige Programme, die mit dem schon oben genannten Rechner hergestellt wurden. Danach kam die erste und immer noch einzige Software für den ATARI ST: MS-DOS-Emulator. Nach langem Warten vieler ATARI-Anwender auf einen Hardware-Emulator (welcher ursprünglich von ATARI selbst angekündigt war), hat eine Firma in Amerika - *Avantgarde Systems* - einen MS-DOS-Emulator für den ATARI ST auf den Markt gebracht. Die Verbreitung des Big-Blues (keine Panik, es handelt sich nicht um einen neu aufgelegten Blues von Modern Talking) und der PC-Kompatiblen hat ohne Zweifel die gesamte Computerwelt stark beeinflußt. Es gibt unter diesem Betriebssystem eine ungeheure Menge an Programmen aller Klassen (gute, weniger gute, schlechte und viele, die man einfach vergessen sollte), die dieses System sehr beliebt machen. Durch den *PC ditto* hat sich für den ATARI ST-Anwender eine neue Welt geöffnet. Ob er aber sinnvoll eingesetzt werden kann, bleibt jedem selbst überlassen. Auf jeden Fall ist für jeden Programmierer, der im Büro vor einem MS-DOS-Rechner sitzt von Vorteil, zu Hause mit einem PC-Kompatiblen arbeiten zu können. Ein anderer und vielleicht noch wichtigerer Punkt sind spezielle Bereiche, wie zum Beispiel beim Cross-Assembler, die aber wegen hoher Entwicklungskosten für den ATARI ST noch nicht existieren oder nicht professionell genug sind

Die Textmaschine

Eine der ersten Software, die es für den ST gab (*Megaroid* ausgenommen), waren Textver-

WIEDER...

Weihnachtsbasar



arbeitungsprogramme wie z.B. *1st_Word*, das auch heute noch existiert. Wegen seiner unkomplizierten Bedienung hat er sich mehr oder weniger als Standard durchgesetzt. Der größte Mangel dieses Programms ist, daß es vom Hersteller wenig geflegt wird. Eine neue Version (V.3.0) soll irgendwann im nächsten Jahr verkauft werden.

Aber auch andere Textsysteme werden für die ST-Serie angeboten. Steve z.B. war nie so einfach zu bedienen wie *1st_Word*, hatte dafür aber eine Menge anderer Funktionen. Von dem Düsseldorfer Softwarehaus wurde eine sinnvolle Anpassung an ein schon existierendes Textverarbeitungssystem entwickelt, das seine Dienste auf dem legendären 64er verrichtet. Von einer Satzsprache, die auf einer großen Rechneranlage arbeitet, und für viele das beste Textverarbeitungssystem für den ATARI ST darstellt, wurde ebenso eine Anpassung gemacht. *TeX* ist auf verschiedenen Rechnern implementiert und wird im naturwissenschaftlichen Bereich sehr oft angewendet. Noch zwei andere Anpassungen schließen die Konstellation von Textsystemen ab: *Star Division* hat Anfang dieses Jahres seinen viel verkauften *Star Writer* für den ST angepaßt und die *WordPerfect Corporation* hat genau das gleiche mit seinem *WordPerfect* getan. Ein neuer Maßstab wurde mit Sicherheit von *SIGNUM* gesetzt. Dieses Textverarbeitungsprogramm - auch eins der ersten - hat unter anderem durch seine Druckqualität und Vielzahl an verfügbaren Zeichensätzen geglänzt.

Der Polyglott

An Programmiersprachen hat es dem ATARI ST nie gefehlt. Am Anfang war es ATARI selbst, die die Programmierer mit einem sehr schlechten BASIC-Interpreter und mit einem schrecklich langsamen LOGO versorgt hat. Für diejenigen, die noch DM 1500,- locker machen konnten, hatte ATARI sogar ein sogenanntes Entwicklungspaket, das aus einem C-Compiler, bzw. einem Assembler bestand. Es dauerte aber nicht lange bis dieser Compiler Konkurrenz bekam. *Application Systems* führte aus Amerika einen wesentlich leistungsfähigeren C-Compiler, das *Megamax C*, ein. *Metacomco* hat seinen bekannten *Lattice C* auf der damals noch neuen Maschine angepaßt. In diesem Jahr hat *Borland* ebenfalls reagiert und eine Version von *Turbo C* an den ATARI ST implementiert.

Nicht anders war oder ist die Entwicklung bei Pascal gewesen. Zuerst hat CCD allein mit seinem Pascal in der ATARI-Szene triumphiert. *Metacomco* brachte später ebenfalls eine eigene Version seines *Pascal-Compilers* auf den Markt. Sogar einen Pascal-Interpreter gibt es mittlerweile für die ST-Serie.

Nur die alten ATARI-Besitzer werden sich noch an diesen schrecklichen BASIC-Interpreter erinnern, der anfangs mit dem ATARI geliefert wurde. Auch heute wird immer noch eine Version dieses verbreiteten Interpreter mitgeliefert. Sein Name lautet jetzt aber anders: *OMIKRON.BASIC*. Es handelt sich aber nicht nur um eine Namensänderung. *OMIKRON.BASIC* existiert seit geraumer Zeit auf dem ATARI-Markt und wurde von der gleichnamigen Firma bis jetzt in der Version 2.0 vertrieben. Dieser Interpreter hat sich bis heute wegen seiner hohen Zuverlässigkeit in vielen Bereichen der Naturwissenschaft etabliert. Die Version 3.0, die von ATARI mitgeliefert wird, hat sich bezüglich ein paar Befehlen geändert, und der Editor wurde völlig neu gestaltet. Der große Sprung in Sachen BASIC-Interpreter wurde aber bereits von Anfang an getan. *GFA-Systemtechnik* zeigte schon auf der CeBIT 1986 die Beta-Version ihres BASIC-Interpreters. Kurz danach kam die vollendete Version auf den Markt. Der *GFA-BASIC-Interpreter* hat neue Maßstäbe in der damaligen Sprachlandschaft des ST gesetzt. Nicht nur daß dieser Interpreter voll funktionsfähig war, er war auch schneller und viel leistungsfähiger. Viele Programmieranfänger hatten auf solch einen Interpreter gewartet. Viele dachten, er würde von ATARI kommen, was aber nicht der Fall war. Dementsprechend war auch die Reaktion. Schon im ersten Jahr wurden 'zigtausend von Interpretern verkauft. Die Begeisterung hat nicht nachgelassen, und mittlerweile gibt es

viele professionelle Programme, die in dieser Sprache geschrieben wurden.

Aber auch bei anderen Sprachen ist die Auswahl groß. Neuerdings ist die zweite Sprache von N.Wirth zur Mode geworden. Mindestens fünf verschiedene *Modula-2 Compiler* konkurrieren miteinander um den ersten Platz. Nicht anders sieht es bei *Forth* aus, wobei eine der mächtigsten Versionen dieser Sprache immer noch die PD-Version ist. *APL* und *PEARL* dagegen existieren als alleinige Vertreter ihrer Rasse. Die 'fünfte Generation' ist ebenfalls mehrmals vertreten. Verschiedene *LISP*- bzw. *Prolog*-Versionen erlauben dem Programmierer, der sich für den Bereich der *Künstlichen Intelligenz* interessiert, seine eigenen Experimente durchführen zu können. Mit Sicherheit haben wir ein paar Exoten vergessen, aber diese Liste sollte kein Beweis der Vollständigkeit sein.

Boticelli, Picasso oder Leonardo - Der ST kann alles

Längst hat der moderne Mensch verstanden, daß für aufwendige Aufgaben der Computer das ideale 'Tier' ist. Aber nicht nur aufwendige und monotone Aufgaben kann der Rechner erledigen, er kann auch die kreativen und künstlerischen meistern. Es gab von Anfang an Programme für den ST, die das Zeichnen auf dem Rechner erlaubten. Niemand, der sich mit dieser Thematik beschäftigt hat, kann *Neochrom* vergessen, das noch heute als Public Domain zu haben ist. Ebensowenig darf man Tom Hudsons legendäres *Degas* und Peter Melzers *STAD* vergessen. *Easy Draw* beschäftigt sich mit der vektororientierten Grafik. CAD-Programme sind ebenso in verschiedenen Versionen, wie in verschiedenen Leistungsstärken für den ST zu bekommen. Desktop Publishing-Software heißt die neue Mode für den ATARI ST. Auch in diesem Bereich gibt es unterschiedliche Programme. Professionelle Versionen sind aber immer noch zu erwarten. Doch es geht sogar noch einen Schritt weiter zum nächsten Trend, dem Desktop-Video, das mit *Imagic* seinen ersten Vertreter auf dem ST gefunden hat.

Daten plus Daten gleich Datenbank

Das Handhaben von Daten aller Art am Computer ist mit Sicherheit neben der Texterfassung eine der häufigsten Anwendungen für Rechner. Auch hier haben die Software-Hersteller die ATARI-Gemeinschaft nicht enttäuscht. Nach anfänglich fast müheloser Anpassung an existierende Datenbänke wurde mit der Produktion maßgeschneideter Datenbänke für den ST richtig losgelegt. Heute existieren verschiedene allgemeine bzw. spezifische Programme in diesem Bereich für die ST-Rechner-Serie. *Adimens*, *Superbase*, *DBASE*, *DB Master One* sind nur ein paar Vertreter einer unzähligen Menge von Datenbankprogrammen, die es für den ST gibt.

Der Softwareberg wächst

Nicht nur Emulatoren, Textverarbeitungsprogramme und Programmiersprachen gibt es in Unmaßen für den ST, sondern auch eine Fülle von anderen nützlichen Programmen. Wenn wir hier nur annähernd versuchen würden, eine komplette Liste von allen Programmen, die zur Zeit für den ATARI ST existieren, aufzustellen, würden wir mit Sicherheit mehr Zeitungsseiten brauchen, als diese Ausgabe hat. Nur bezüglich der Betriebssysteme gibt es außer TOS noch vier andere, die erwähnenswert sind. Für fast jede Branche existiert eine spezifische Softwarelösung. Editoren und andere Utilities werden von renommierten und weniger renommierten Softwarehäusern angeboten. Wie gesagt, diese hier alle zu nennen, würde den Rahmen dieses Heftes sprengen.

Zweite Etage: Hardware

Was braucht man außer dem Rechner selbst und ein wenig Software? Diese Frage könnte unter Umständen als sehr vernünftig angesehen werden. Aber wenn man es sich genau überlegt, ist der nackte Rechner eben mehr als das:

Fangen wir mit dem internen Speichermedium an. Die anfangs gelieferten ST-Rechner waren 'nur' mit einem halben Megabyte ausgestattet. Deshalb sind nach der Einführung des ST-Rechners die Anbieter von Speicherausrüstung wie Pilze aus dem Boden geschossen. Mit der neuen Generation von ST-Rechnern ist dieses Problem mittlerweile gelöst. Aber gehen wir weiter zu dem externen Massenspeicher. Das größte Paradoxon, das sich ATARI bis jetzt geleistet hat, war, daß mit dem allerersten Rechner eine Floppy mitgeliefert wurde, die nicht in der Lage war, die gesamte Kapazität des Rechners auf Diskette zu schreiben. Es hat sich wieder eine Lücke geöffnet, die von zahlreichen Anbietern (selbst von ATARI) genutzt wurde, um größere und teilweise bessere Laufwerke anzubieten. Neben diesem Angebot wurden zusätzlich *5 $^1/_4$"-Floppies* angeboten, die man mühelos an den Rechner anschließen konnte. Aber nicht nur von Floppies lebt der Mensch. Auch Festplatten waren von Anfang an sehr gefragt. Neben ATARI bieten jetzt viele Hardware-Hersteller Festplatten in sehr unterschiedlicher Größe für den ATARI an.

Für diejenigen, die viel Wert auf Datensicherung legen, ist ein *Streamer* nicht nur ein sinnvolles, sondern auch ein notwendiges Gerät. Genauso ist eine auswechselbare Platte (wenn sie nicht so teuer wäre) ein qualitativer Ersatz für eine Festplatte. Wenn man gern die Bibel auf dem ATARI ST lesen möchte, dann braucht man unbedingt ein *CD-ROM*. Die Software dafür haben schon Claus Brod und Anton Stepper auf der Düsseldorfer Messe gezeigt.

Aber für alle ATARI-Anwender, die nicht viel mit Platten zu tun haben möchten, gibt es in anderen Bereichen viele interessante Ergänzungen zu ihrem Rechner. Bitte folgen Sie mir, ich werde Ihnen eine wunderbare Welt zeigen.

Digitizer, Scanner & andere Apparate

Wie wäre es mit einer Art 'Auge' für den ST? Nicht schlecht! Da könnte man seinen Rechner für alle möglichen observatorischen Versuche mißbrauchen (es gibt auch in der Naturwissenschaft wertvolle Vorgänge, die man sich durchaus anschauen kann. Also nicht, was Sie gedacht haben). Hier würde ein Digitalisiergerät wunderbar passen. Und solch ein Gerät, ein *Digitizer*, wird in verschiedenen Auflösungen und Qualitäten von sehr unterschiedlichen Firmen angeboten. Da wir bei Aufnahmen sind, betrachten wir die Sache von einer anderen Sicht aus. Stellen Sie sich einmal vor, Sie hätten eine Bildvorlage (irgendein Buch oder einen Zeitungsausschnitt), die Sie gerne in einen Text einbinden möchten. Mit einem Digitizer würde die Sache ziemlich kompliziert, das geeignete Gerät dafür wäre ein *Scanner*. Und auch hier ist das Angebot groß genug.

NEU & AKTUELL



Die neuartige Einführung in die Textverarbeitung, die verspricht ein Klassiker zu werden. Ein Buch, das man auch einmal am Wochenende lesen möchte – begeisternd, und manchmal auch bissig – über ein „Programm", das nie ganz ausgereizt wurde.

MERKMALE:

- ▶ Voll von Information, die menschlich verpackt ist
- ▶ Zeigt minutiös die Technik der Textverarbeitung, ohne ihre Faszination zu vergessen
- ▶ Vom Praktiker für Praktiker: Für Autoren und Ärzte... Sekretärinnen und Schwiegermütter... Journalisten und Geschäftsleute
- ▶ Für Anfänger, die schnell lernen wollen und nichts vom Computer verstehen
- ▶ Für Fortgeschrittene, die sich darüber hinaus für RAM-Disk, Zeit sparen, Grafik und Drucklegung interessieren
- ▶ Für Kenner, die mal sehen wollen, ob es bei der Textverarbeitung noch etwas gibt, das sie nicht wissen
- ▶ Mit vielen Beispielen und Grafiken auf über 300 Seiten

INHALT:

- ▶ Wie man sofort Text auf den Bildschirm bringt
- ▶ Der unendliche Zeichensatz und 1st Proportional
- ▶ Alle Menüs auf einen Blick
- ▶ Tippen, Korrigieren, Zeitsparen, Wörterbuch...
- ▶ Formatieren und Umbruch
- ▶ Einbinden von Grafik, DEGAS im Text, selber malen, drucken
- ▶ Ist eine Art DTP möglich?
- ▶ 1ST MAIL – auf 30 Seiten so erklärt, daß es jeder versteht
- ▶ Zum ersten Mal: Vom Manuskript zum Buchdruck!
- ▶ Was man nicht im Handbuch findet: 100 Tips u. Tricks
- ▶ Verschiedene Bildgrößen
- ▶ Wie man den Treiber zum Großschreiben bringt
- ▶ „Was tun wenn...?" Eine lange Liste von Ratschlägen
- ▶ Der Umgang mit dem Drucker – Dichte, Schnelligkeit, Zeichensatz, Papiereinzug, Berechnung von Kopf- und Fußzeile
- ▶ Übersetzung der wichtigsten Fachausdrücke
- ▶ „1ST WORD TRAINER" – Aufgaben, Probleme, Fragen: Ein Übungsteil, der zum Denken auffordert.

Über 300 Seiten B-414 **DM 49,-**




MERKMALE:

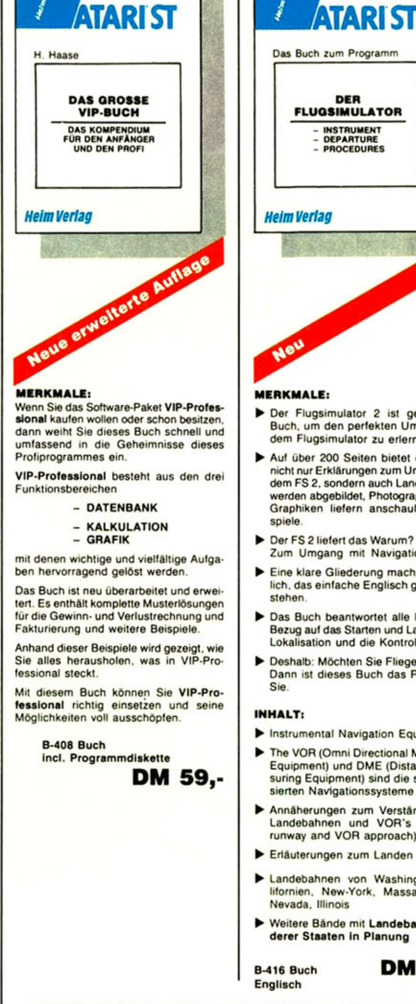
Der Grundlehrgang ist das Buch für den richtigen Einstieg mit dem ATARI ST. Auf über 450 Seiten wird der Leser leicht verständlich in die Bedienung des Rechners eingeführt. Die vorliegende dritte Auflage wurde von Grund auf überarbeitet und stark erweitert. Sie berücksichtigt nun alle neuen Rechner (inclusive MEGA ST) und bietet eine gründliche Einführung in BASIC. Auf der beiliegenden Programmdiskette sind alle Beispielprogramme enthalten. Für Festplattenbesitzer ist eine Beschreibung der ATARI-Harddisk-Treibersoftware sowie ihres Einsatzes enthalten. Auf der Programmdiskette ist außerdem ein Back-up-Programm gespeichert.
Ein Utility-Kapitel beschreibt die auf der Diskette befindlichen Hilfsprogramme, die eine Grundausstattung jedes Software-Pools darstellen.
Professionelleren Ansprüchen genügen die Programme, die Ihnen in einem weiteren Abschnitt vorgestellt werden. Ihre Vor- (und Nach)teile erfahren Sie in kurzen Beschreibungen der Software.

INHALT:

- ▶ Entwicklungsgeschichte
- ▶ Die Hardware
- ▶ Das Betriebssystem
- ▶ Arbeiten mit der Festplatte
- ▶ ST-BASIC
- ▶ Nützliche Hilfsprogramme **auf Diskette**
- ▶ Vorstellung ausgewählter Programme
  - Signum!2
  - ST-Pascal Plus
  - Omikron-BASIC
  - GFA-BASIC
  - Tempus
- ▶ Anhänge
  - Index
  - Worterklärungen
  - Tabellen
  - Literaturverzeichnis

Hardcover, 453 Seiten mit Programmdiskette
B-400 **DM 59,-**

Alle Preise sind unverbindlich empfohlene Verkaufspreise.



MERKMALE:

Wenn Sie das Software-Paket **VIP-Professional** kaufen wollen oder schon besitzen, dann weist Sie dieses Buch schnell und umfassend in die Geheimnisse dieses Profiprogrammes ein.

VIP-Professional besteht aus den drei Funktionsbereichen

- **DATENBANK**
- **KALKULATION**
- **GRAFIK**

mit denen wichtige und vielfältige Aufgaben hervorragend gelöst werden.

Das Buch ist neu überarbeitet und erweitert. Es enthält komplette Musterlösungen für die Gewinn- und Verlustrechnung und Fakturierung und weitere Beispiele.

Anhand dieser Beispiele wird gezeigt, wie Sie alles herausholen, was in VIP-Professional steckt.

Mit diesem Buch können Sie **VIP-Professional** richtig einsetzen und seine Möglichkeiten voll ausschöpfen.

B-408 Buch incl. Programmdiskette **DM 59,-**



MERKMALE:

- ▶ Der Flugsimulator 2 ist genau das Buch, um den perfekten Umgang mit dem Flugsimulator zu erlernen.
- ▶ Auf über 200 Seiten bietet das Buch nicht nur Erklärungen zum Umgang mit dem FS 2, sondern auch Landebahnen werden abgebildet, Photographien und Graphiken liefern anschauliche Beispiele.
- ▶ Der FS 2 liefert das Warum? und Wie? Zum Umgang mit Navigation u.v.m.
- ▶ Eine klare Gliederung macht es möglich, das einfache Englisch gut zu verstehen.
- ▶ Das Buch beantwortet alle Fragen in Bezug auf das Starten und Landen, die Lokalisation und die Kontrolle.
- ▶ Deshalb: Möchten Sie Fliegen lernen? Dann ist dieses Buch das Richtig für Sie.

INHALT:

- ▶ Instrumental Navigation Equipment
- ▶ The VOR (Omni Directional Measuring Equipment) und DME (Distance Measuring Equipment) sind die standardisierten Navigationssysteme
- ▶ Annäherungen zum Verständnis von Landebahnen und VOR's (genannt runway and VOR approach)
- ▶ Erläuterungen zum Landen
- ▶ Landebahnen von Washington, Californien, New-York, Massachusetts, Nevada, Illinois
- ▶ Weitere Bände mit **Landebahnen anderer Staaten in Planung**

B-416 Buch Englisch **DM 49,-**

BESTELLCOUPON

einsenden an
Heim Verlag · Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt

Ich bestelle: ____ St. 1ST WORD PLUS à 49,– DM
____ St. Grundlehrgang mit Programm-Diskette à 59,– DM
____ St. Das große VIP-Buch mit Programm-Diskette à 59,– DM
____ St. Der Flugsimulator (Buch zum Programm, engl.) à 49,– DM

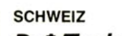
zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.



Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

SCHWEIZ
DataTrade AG
Langstr. 94
CH-8021 Zürich

OMIKRON. Basic

NEU 3.0

DIE SPITZEN-BÜCHER ZUM NEUEN ATARI-BASIC

ATARI ST – Schneider/Steinmeier – PROGRAMMIEREN IN OMIKRON BASIC – Heim-Verlag

MERKMALE:

▶ Omikron-BASIC 3.0 – das neue Standard-BASIC für den ATARI ST – wird immer häufiger zur Erstellung von großen u. kleinen Programmen eingesetzt. ▶ Das Buch „Programmieren in Omikron-BASIC" zeigt an vielen Beispielen die Entwicklung von Programmen. Anhand zahlreicher dokumentierter Listings wird der Umgang mit den besonderen Elementen dieser BASIC-Version erläutert. ▶ In einem 1. Abschnitt werden wichtige Unter- u. Hilfsprogramme vorgestellt: Druckeranpassung, universelle Zahlensystemumwandlung, Feststellen der Existenz einer Diskettendatei, Programmierung grafischer Blockoperationen mit dem BITBLT-Befehl u.a.m. ▶ Ein weiteres Kapitel behandelt den Einsatz von GEM-Funktionen in Omikron-BASIC-Programmen: Der Umgang mit Fenstern, Dialogboxen, der Fileselectbox u.a.m. Auch die Verwendung der zum BASIC gehörenden GEMLIB wird behandelt. ▶ Um die Programmierung zu erleichtern, werden einige Hilfsprogramme gezeigt, die z. B. das Kopieren mehrerer Files vereinfachen oder eine Datei in DATA-Zeilen umwandeln. Auch ein Editor für Mauszeiger u. einige Füllmuster fehlen nicht. ▶ Die grafischen Fähigkeiten des Computers u. die entsprechenden Anweisungen des BASIC-Interpreters werden anhand der Entwicklung einer dreidimensionalen Grafik mit Rotation u. Projektion sowie einer Turtle-Grafik u. weiteren kleinen Beispielen demonstriert. ▶ 4 größere Anwendungen bieten die Gelegenheit, sich mit umfangreicheren Projekten vertraut zu machen. ▶ Für Schüler u. interessierte Laien sind einige Anwendungen aus dem mathematischen Bereich gedacht. ▶ Zur Abrundung u. Auflockerung enthält das Buch einige Spielprogramme.

INHALT:

▶ Druckeranpassung ▶ Feststellen der Existenz einer Datei ▶ Das Verwenden mehrerer Bildschirme ▶ Einbinden von Funktionen in laufende Programme ▶ GEM-Programmierung ▶ Arbeiten mit Fenstern, Dialogboxen u. Menüzeilen ▶ Punktgenaue Eingabe auf dem Bildschirm ▶ Komfortable Filenameneingabe mit Fileselect ▶ Programmierung der Alertboxen ▶ Hilfsprogramme ▶ Editor für Mauszeiger u. Füllmuster ▶ Kopieren von mehreren Files ▶ Dreidimensionale Darstellung von Funktionen ▶ Turtle-Grafik ▶ Adreßverwaltung ▶ Ermitteln von Mittelwerten, Varianz u. Standardabweichungen ▶ Integration nach Simpson ▶ Spiele

ca. 400 Seiten **DM 49,-** Best.-Nr. B-411

Diskette mit allen abgedruckten Programmen **DM 39,-** Best.-Nr. D-431

ATARI ST – Schneider/Steinmeier – kurz & klar OMIKRON BASIC – Ein fundiertes Nachschlagewerk – Heim-Verlag

INHALT:

ATARI hat sich entschlossen, endlich eine wirklich leistungsfähige Programmiersprache mit den Rechnern der ST-Serie auszuliefern. Daß die Wahl gerade auf Omikron-BASIC fiel, ist kein Wunder, denn diese Sprache ist nicht nur besonders einfach zu erlernen, sondern stellt zugleich einen Leistungsumfang zur Verfügung, der selbst das Schreiben professioneller Anwendungen erlaubt. Um die über 200 Befehle mit ihren zahlreichen Parametern nutzen zu können, ist eine alphabetische Übersicht der Kommandos und ihrer Möglichkeiten unverzichtbar, und genau dies liefert „Kurz & Klar Omikron-BASIC 3.0".
Sie erfahren, mit welchen Anweisungen man Matrizen multipliziert, invertiert oder eine Determinante ermittelt, wie man Linien und Kreise zeichnet oder mit welchem Befehl man eine Alertbox auf den Bildschirm bringt. Außer der reinen Befehlsübersicht, die an sich schon eine enorme Hilfe bei der Programmentwicklung darstellt, geben die zahlreichen Anhänge Auskunft über die Bedeutung der Modi des BITBLT-Befehls, die ASCII-Zeichen des ST's oder die Zuordnung der Tasten zu den Scan-Codes der INKEY$-Funktion. Auch die VT52-Codes werden in einem Anhang behandelt.
Wenn man einen Befehl aus einem bestimmten Anwendungsgebiet sucht, hilft der nach Anwendungen geordnete Befehlsindex weiter. Dem Besitzer eines Compilers, der für das Erstellen eigenständiger Programme erforderlich ist, hilft ein Abshcnitt über die Compilerdirektiven und andere Besonderheiten weiter.
Die neueste Auflage dieses praktischen Hardcover-gebundenen Nachschlagewerkes berücksichtigt selbstverständlich die neue Version 3.0 des Omikron-BASICs.

Über 200 Seiten **DM 29,-** Best.-Nr. B-412

ATARI ST – DAS GROSSE OMIKRON. 3.0 BUCH – Heim Verlag

MERKMALE:

▶ Omikron-BASIC 3.0 ist der neue BASIC-Standard für den ATARI ST. Das vielfach bewährte „große Omikron-BASIC 3.0-Buch" gibt es nun in einer neuen Auflage, die alle Neuheiten berücksichtigt. ▶ Das Buch stellt einen leicht verständlichen Einstieg für den Anfänger in die Programmierung einer der leistungsfähigsten Sprachen für den ATARI ST dar. ▶ Jeder, der sich ernsthaft mit den Möglichkeiten dieser Programmiersprache beschäftigen möchte, benötigt dieses fundierte Lehrbuch. Aber auch der Umsteiger von einem anderen BASIC-Dialekt findet hier alle notwendigen Informationen, um mit Omikron-BASIC optimal arbeiten zu können. ▶ Dem Einsteiger bietet „Das große Omikron-BASIC 3.0-Buch" eine systematische und leicht verständliche Einführung, die von den Schleifen- und Programmstrukturen über die unterschiedlichen Variablentypen und die Arbeit mit Feldern bis zu den numerischen und Stringfunktionen reicht. Weiterhin findet der Leser Hinweise zum Umgang mit Unterprogrammen und Prozeduren. Aufbauend auf diesen Teil ist die Programmierung der Multitasking-Fähigkeiten anhand eines Drucker-Spoolers sowie der Einsatz von die Arbeit erleichternden abstrakten Datentypen erklärt. Weitere Abschnitte beschäftigen sich mit Dateien sowie mit der Programmierung von Grafik und Betriebssystemfunktionen. Auf diese Weise wird dem Neuling vom ersten Einzeller bis zu komplexen Programmen die Arbeit mit diesem komfortablen BASIC-Dialekt nahegebracht. ▶ Viele Beispielprogramme, die teilweise, wie etwa ein Fakturierungsprogramm, explizit entwickelt werden, runden das Buch ab. Damit die einzelnen Programme verständlicher werden, sind sie, wo nötig, durch Ablaufdiagramme ergänzt. Um nicht alle Listings abtippen zu müssen, liegt dem Buch eine Diskette mit allen Programmen bei. ▶ Aber auch, wenn man BASIC bereits beherrscht, ist das Buch durch seine zahlreichen Anhänge, die unter anderem eine ASCII-Tabelle, eine Übersicht der Füllmuster und BITBLT-Modi sowie ein Verzeichnis der VT52-Codes und einen Index enthalten, als Nachschlagewerk wertvoll. Zu diesem Zweck wurde auch eine vollständige alphabetisch sortierte Kurzübersicht der Befehle integriert.

INHALT:

▶ Erklärung der Schleifen- und Programmstrukturen – *Primzahlenberechnung – Zahlenraten* ▶ Variablentypen und Arrays – *Sieb des Eratosthenes – Adreßeingabe* ▶ Numerische und Stringfunktionen – *Alle trigonometrische Funktionen wie Sinus, Cosinus etc. – Die Zufallsfunktion* ▶ Unterprogramme und Prozeduren – *Rekursive Prozeduraufrufe – Suche des Ausgangs aus einem Labyrinth* ▶ Grafikprogrammierung – *Grafische Grundelemente wie Linie, Kreis, Rechteck – Blockoperationen – Flimmerfreie Animation durch Einsatz mehrerer Bildschirme – Erzeugen eines Balkendiagramms* ▶ Multitasking in Omikron-BASIC – *Druckerspooler* ▶ Programmierung von Abstrakten Datentypen – *Die Datenstruktur „Schlange" – Verkettete Listen* ▶ Dateiverwaltung – *Programmierung von sequentiellen Dateien – Arbeiten mit Random-Access-Dateien* ▶ Programmprojekt Fakturierung – *Ausführliche Entwicklung eines Programms auf ca. 30 Seiten* ▶ Betriebssystemprogrammierung – *Einsatz von Alertboxen und der Fileselectbox – Benutzen von Pulldown-Menüs, Fenstern und Dialogboxen – Programmierung einer Druckeranpassung in BASIC* ▶ Programmentwicklung und Debugging – *Fehlersuche und -beseitigung* ▶ Sammlung von Beispielprogrammen – *Ausgabe eines Diskettenverzeichnisses – Backup-Programm für die Harddisk – Turtle-Grafik in Omikron-BASIC – Adreßverwaltung* ▶ Alphabetische Befehlsübersicht

Hardcover
ca. 400 Seiten einschl. Programm-Diskette **DM 59,–** Best.-Nr. B-413

Preise sind unverbindlich empfohlene Verkaufspreise

BESTELLCOUPON

einsenden an
Heim Verlag · Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt

Ich bestelle: ____ St. Das große OMIKRON-BASIC-BUCH (incl. Programm-Diskette) à DM 59,-
____ St. Programmieren in OMIKRON-BASIC à DM 49,-
____ St. PROGRAMMDISKETTE zum Buch à DM 39,-
____ St. Kurz & Klar Nachschlagewerk OMIKRON-BASIC à DM 29,-

zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

SCHWEIZ
DataTrade AG
Langstr. 94
CH-8021 Zürich

Mein Rechner bekommt Anschluß

Nicht nur im Aufnahmebereich stellen viele Firmen Zubehör für den ST her. Genauso sinnvoll sind *Coprozessoren*, die die Leistung des ST-Rechners um einiges steigern, oder Umrüstungsplatinen die die Aufnahme eines anderen Prozessors ermöglichen.

Haben Sie mit Meßauswertung bzw. -erfassung zu tun? Ja? Na prima! Dann brauchen Sie mit Sicherheit eine Karte, die zuerst die Erfassung möglich macht. Oder möchten Sie mit dem Computer etwas steuern? Vielleicht die Heizung oder einen Roboterarm oder...? Dann brauchen Sie auch ein Interface, das die Daten vom Computer zum zu steuernden Gerät umleitet.

Beklagt sich Ihre Maus über die raue Oberfläche ihres Tisches, oder hat sie nachts Schüttelfrost? Dann ist sie ein klarer Fall für ein *Mouse-Pad* und einen *Mouse-Mantel*. Möchte man den ATARI ST an einem Schwarzweiß- und an einem Farbmonitor bedienen, dann benötigt man einen *Umschalter*. Einen *HF-Modulator* braucht man, wenn der ST statt an einem Monitor, an einem konventionellen Farbfernsehgerät (das gibt es auch) betrieben wird. Auch diesen gibt es massenweise zu kaufen.

Wie Sie sehen, ist mit der Anschaffung des Rechners alleine nicht alles getan. Man braucht unter Umständen eine ganze Menge zusätzlicher Sachen. Insofern war die Frage am Anfang gar nicht so unberechtigt.

Letzte Etage: Bücher

Mag sein, daß der Mensch ein 'Spielwesen' ist, aber sein Drang, alles auf Papier festzuhalten, ist grenzenlos. Seit Gutenbergs revolutionärer Erfindung, dem Buchdruck, hat die Schriftsprache unvorstellbare Dimensionen angenommen. Nicht nur die allgemeine Literatur profitiert davon, sondern auch Fachbücher. Die Computerbücher haben sich in den lezten 10 Jahren ebenso vermehrt. Auch ATARI ST hat von diesem Sturm etwas abbekommen. 'Zig Bücher, Tonnen von Papier und tausende Worte mit unzähligen Informationen wurden für den ST abgedruckt. Ob alles gut ist? Das möchten wir bezweifeln. Aber auch hier ist unsere Meinung sehr einseitig und subjektiv.

Als der ST noch jung war, brachte *Data-Becker-Verlag* zwei Bücher heraus, die mit großer Sicherheit für jeden ST-Programmierer eine enorme Hilfe bedeuteten. Das *ST-Intern* und das *GEM-Buch* waren die ersten deutschsprachigen Dokumentationen über das Innenleben des ATARI ST-Rechners. Mit der immer schnelleren und größeren Verbreitung des ST, wuchs auch das Interesse für Fachliteratur in vielen unterschiedlichen Bereichen. Heutzutage gibt es nicht nur ein Buch für ein spezifisches Thema, sondern gleich mehrere. Die Auswahl ist groß, die Qual noch größer.

Eins der besten Bücher für den ATARI ST ist gar nicht so alt. Das *Profi-Buch* vom *Sybex-Verlag* ist eine sehr erwähnenswerte Ausnahme in der Literaturlandschaft. Es enthält fast alle notwendigen Informationen für den Auf-, aber auch für den Einsteiger. Einfach ein Muß für jeden ATARI ST-Anwender, der sich ernsthaft mit seinem Rechner beschäftigen möchte.

Für den Programmierer, der an Massenspeichern Interesse hat, ist *Scheibenkleister* von Claus Brod und Anton Stepper zu empfehlen. Dieses Buch ist ein fundiertes Nachschlagewerk für die Massenspeicher, die für den Atari auf dem Markt sind. Außerdem wurde es in einem für Computerbücher außergewöhnlich humorvollen Stil geschrieben.
Wir hatten schon erwähnt, daß es eine Unmenge von verschiedenen Programmiersprachen für den ATARI ST gibt. Mittlerweile existieren genausoviele Lehrbücher für diese Sprachen. Der Münchener Verlag *Markt & Technik* brachte mit seinem *Assembler-Buch* von Peter Wollschläger ein durchaus interessantes Buch über ein meist trockenes und schwieriges Thema heraus. Sehr beliebte Sprachen wie GFA-BASIC zum Beispiel werden mit einer Fülle von Publikationen honoriert.

Grafik und Sound sind genauso zahlreich vertreten wie das Programmieren der geliebten GEM-Oberfläche. Auch Bücher über Anwendungen, wie Textverarbeitung oder Tabellen-Kalkulation, helfen dem Anwender beim Einstieg in diese Bereiche. Alles in allem ist der ATARI ST-Anwender nicht allein mit seinen Problemen. Fast zu jedem Thema kann er hier und da ein wenig Information aufschnappen. Mit ein wenig Geduld und vielleicht viel Geld, wird sich jeder die richtige Bibliothek für seinen ATARI anschaffen.

Die folgende Liste ist natürlich keine vollständige Übersicht über alle Produkte, die es für den ATARI ST gibt; wir haben es aber versucht, Ihnen eine kleine Auswahl vorzustellen. Es soll auch keine Qualitätsauswahl sein, da wir bestimmt das ein oder andere wirklich gute Produkt vergessen bzw. nicht aufgeführt haben, da wir sonst den Rahmen dieser Ausgabe gesprengt hätten.

MM

SOFTWARE

Pro.Name	Vertrieb	Preis	Heft
Emulatoren und andere Betriebssysteme			
Aladin 2.1	Softpaquet	ab 398,-	11/88
Eumel	Bios Computer GmbH	285,-	8/87
Kat-Ce-ST	Christa Mayer-Gürr	100,-	7/88
MIRAGE	gdat	348,-	8-9/88
OS-9		~1500,-	5/87
PC ditto	MAXON Computer	198,-	12/87
RTOS	Heise-Verlag	248,-	4/87
CAD- und Platinenlayout-Programme			
CAD Project	Schwarzstein Software	ab 298,-	3/88
Campus	Technobox Software GmbH	798,-	6/87
Campus Draft	Technobox Software GmbH	149,-	7/88
First CAD	PublicSoft	250,-	11/87
GFA-Draft+	GFA-Systemtechnik	349,-	10/87
PCB Layout	Thomas Praefcke	199,50	2/88
Platine ST	Data Becker	99,-	5/87

SOFTWARE

Pro.Name	Vertrieb	Preis	Heft
Datenbank- und Adreßverwaltungs-Programme			
Adimens	ATARI, ADI Software	199,-	11/87
Aditalk	ATARI, ADI Software	199,-	3/88
Adimens Prog	Markt & Technik	199,-	11/88
Search	pcp	69,-	12/87
Superbase	Markt & Technik	249,-	12/87
Themadat	Try Soft	49,-	12/87
dBase ST	Markt & Technik	348,-	11/87
RegentBase	H.Haase	249,-	11/87
Textverarbeitungen und Editoren			
2nd_Word	MAXON Computer	59,-	
Beckertext	Data Becker	199,-	6/87
Signum2	Application Systems	448,-	12/87
Star Writer 1.01	Star-Division GmbH	198,-	6/88
STEVE	Computertech. Kiekbusch	348,-	11/87
TeX	Kettler EDV-Consulting	395,-	8/87
Tempus 2.0	CCD	129,-	10/88
WordPerfect	WordPerfect	ca. 800,-	10/88
WordPlus2.02	ATARI	199,-	3/88

Liste geht weiter...

SOFTWARE

Pro.Name	Vertrieb	Preis	Heft
Mal- und Zeichen-Programme			
Degas	Computertech. Kiekbusch	169,-	5/86
Easy Draw	Computertech. Kiekbusch	448,-	5/86
Imagic Grafik Comp.	Application Systems	478,-	3/88
Megapaint	TommySoftware	298,-	
NEOCrome	Public Domain	PD	5/86
Panip	Adventure Soft	124,5	6/87
Printmaster	G.Knupe	179,-	9/86
STAD	Application Systems	179,-	2/87
Desktop Publisher			
Calamus	ATARI	398,-	12/88
Timeworks	Computertech. Kiekbusch	398,-	8/88
Publishing-Partner	G. Knupe	498,-	4/88
Musik			
EZ-Score Plus	Hybrid Arts	199,-	6/88
Superscore	Bela Computer	698,-	8-9/88
Twenty Four	Steinberg Research	390,-	10/86
Musix32	TommySoftware	99,-	4/87
K-Minstrel	Knupe	98,50	4/87
Pro Sound Designer	CCD	165,-	1/88
AS Sound	G Data	145,-	1/88
ST Replay	Print Technik Knupe	298,-	1/88
Programmiersprachen			
APL			
APL 68000	gdat	298,-	10/87
Assembler			
Ideal	OMIKRON	99,-	11/86
GFA-Assembler	GFA-Systemtechnik	149,-	11/88
GST-ASM	ATARI	149,-	11/86
Profimat	Data Becker	99,-	2/87
K-Seka	Knupe	169,-	2/86
BASIC			
GFA-Basic 3.0	GFA-Systemtechnik	198,-	5/88
OMIKRON.BASIC 3.00	ATARI	19,90	
C			
Laser C	Application Systems	398,-	8-9/88
Lattice C	Knupe	298,-	10/87
Turbo C	Heimsoeth & Borland	198,-	9/88
Mark Williams	Philgerma	348,-	2/87
FORTRAN			
Prospero-Fortran	Plünecke	317,-	2/88
Modula-2			
Jefferson Modula	KV in Deutschland	$50,-	4/88
Megamax Modula 2	Application Systems	398,-	5/88
Modula 2	SPC	349,-	6/88
Modula 2	Softwave	199,-	6/88
TDI-Modula	Philgerma	248,-	6/88
Pascal			
Pascal+	CCD		
ProPascal	Plünecke	237,-	5/87
Mcc-Pascal	Philgerma	248,-	3/87
Alice Pascal	Looking Glass	$79,95	10/87
Prolog			
Salix Prolog	Brainware	198,-	6/87
TOY-Prolog	Public Domain	PD	5/87
MProlog	Epsilon	1498,-	3/87
LISP			
Cambridge Lisp	Metacomco	490,-	1/87
XLisp	MAXON Computer	PD	1/87
lispas	Tommy Software	198,-	6/87
DFÜ			
Ansiterm	Computerware Sender	89,-	4/88
BTX-Manager	Drews EDV+BTX	428,-	11/88
Interlink	Bela Computer	79,-	8-9/88
Utilities			
Flexdisk	Application Systemes	69,-	12/87
K-Switch	Knupe	98,-	3/87
MT C-Shell	Computerware Sender	298,-	7/88
HD Accelerator	Computerware Sender	98,-	1/88
HD & Extension	GDATA	129,-	1/88
Harddisk Utility	Application Systems	69,-	1/88
1st_Speeder	TommySoftware	89,-	2/88
Finanz- und Kalkulationsprogramme			
Logistix	Philgerma	ca. 500,-	8/87
K-Spread	Knupe	198,-	7/86
ST-Kontor	Sybex	498,-	7-9/88
TIM	C.A.S.H.	198,-	8/87
VIP Professional	Computertech. Kieckbusch	848,-	3/87
BS-Fibu	Bavaria Soft	1149,-	4/87
Bank-X-Fer	C.A.S.H	298,-	10/88
Cashflow	C.A.S.H.	298,-	5/88
fibuMAN	Novoplan	ab 395,-	12/88
V_Manager	Hyper Soft	1498,-	1/88

HARDWARE

Pro.Name	Vertrieb	Preis	Heft
Drucker			
C-715A	C.Itoh	3298,-	7/88
HL-8	Brother	5250,-	11/88
HQP-45	Citizen	2298,-	7/88
LQ 500	Epson	1098,-	4/88
M-1724L	Brother	1995,-	7/88
Microline 390	OKI	~2000,-	11/88
Microline 393C	OKI	3219,-	7/88
NB24-15	Star	2500,-	12/88
P2200	NEC	950,-	12/88
P6-plus	NEC	~2000,-	11/88
PL-450	nbn Elektronik GmbH	2622,-	10/88
SL-130 AI	Seikosha	1899,-	7/88
SLM804	ATARI	3498,-	12/88
Coprozessoren / Transputer			
MC 68881	Lischka	729,-	11/88
ATARI SFP004 68881	ATARI	398,-	
KMAX	Kuma	~3000,-	9/87
PAK-68	ESD	~1098,-	11/88
Speichererweiterung			
Speicherweiterung	ECKL	ab 998,-	9/87
Speicherweiterung	Rhothron	ab 898,-	9/87
Speicherweiterung	Weide	ab 600,-	9/87
Grafikerweiterungen / Scanner/ Digitizer			
Omega Grafiksystem	Omega	~3000,-	1/88
Digitizer Tablett	CRP	ab 999,-	2/88
CP 14 Scanner	Hawk	~3000,-	
Panasonic Scanner	Computerstudio Brock	3598,-	
Easytizer	MAXON Computer	289,-	10/87
Speichermedien und EPROM-Brenner			
CD-ROM	ATARI		
EX 20	Eickmann Computer	1098,-	
HDplus 20	Vortex	1098,-	1/88
SH 205	ATARI	999,-	12/88
Streamer	Rhothron	ab 2100,-	8-9/88
Wechselplatte	Rhothron	4498,-	10/88
Wechselplatte	3K	3298,-	
Juniorprommer	MAXON Computer	198,-	2/88
Profibank	MAXON Computer	ab 348,-	10/88
Sounddigitalisierer und Sprachausgabe			
Pro Sound	CCD	165,-	1/88
AS Sound	G Data	249,-	1/88
ST Replay	Print Technik	298,-	1/88
Sprachausgabe	Schlegel Datentechnik	~300,-	8/87
Allgemeines, Interfaces und ATARI-Nachbauten			
Multibuffer	Elite	199,-	4/88
ST-Oszillographen	Microcomputer-Labor	448,-	3/87
Transfile ST	Yellow Computing	99,-	8/87
A-NET	DM Computer GmbH	ab 1450,-	10/87
SM124 Multisync	Eickmann Computer	698,-	

Liste geht weiter...



Adalbertstr.44 - 5100 Aachen - 0241/24252

1st PROPORTIONAL

siehe Testbericht ST-Computer 6/87

1st PROPORTIONAL ermöglicht den Ausdruck von 1st Word Plus Texten in PROPORTIONALSCHRIFT im BLOCKSATZ! Unterstützt alle proportionalschrift-fähigen Drucker (9, 24 Nadeln) und Typenraddrucker mit PS-Typenrad sowie KYOCERA-Laser.

- läuft vollständig unter GEM (SW-Monitor)
- unterstützt alle Optionen von 1st Word Plus
- verschiedene Zeilenlineale in Proportionalschrift im Blocksatz
- spezielle Grafiktreiber für 9, 18, 24 Nadeldrucker, dadurch endlich satte Schwärzen auch auf 24 Nadeldrucker sowie unverzerrter Ausdruck
- Lieferumfang: Diskette, 60 seitiges Handbuch
- Druckertreiber für NEC P2200, P5/6/7, EPSON LQ 500, 800, 850, FX 85, SEIKOSHA SL 80 AI, STAR NL 10 m.par.Interface, BROTHER HR 15, 20, 25, 35, TA GABI 9009, KYOCERA Laser F1010, 1100,1200...
- 6 seitiges Info mit Probeausdrucken anfordern (2DM)

1st PROPORTIONAL	95.-
SIGNUM II	369.-
TEMPUS V2.0	95.-
CALAMUS DTP	380.-
WORDPLUS	185.-
MEGAMAX MODULA	329.-
DEVPAC ASS. V2.0	137.-
GFA ASSEMBLER	137.-
STAD	154.-
PUBLIC PAINTER	135.-
SUPERBASE PROF.	589.-
ADIPROG/ADITALK	177.-
fibuMAN f V3.0	725.-
fibuSTAT V2.3	398.-

unverb. GESAMTKATALOG anfordern !

PROFI TEXT MODUL

für 1st Word Plus!

Bedienung und Sonderzeichen schnell und einfach per Tastatur • 0,75 sowie 1,5 Zeilenabstand und Sonderzeichen für Atari-Laser • Neue komfortable Fenstertechnik • Vor- und Zurückblättern, Textanfang/ -ende per Tastendruck • Volle Bildschirmgröße für Text • Und vieles mehr! **Nur 98,- DM.** (zuzüglich 5 DM Versandkosten)

Ausführlicher Test-Bericht in ST-Computer 11/88. Lieferung nur gegen Scheck od. per Nachnahme.

Software Service J. Schampel
Landsbergerstraße 87/IV
8 München 2, Tel. 089/5021265

Wo? Natürlich bei Ihrem Atari-Händler!
oder direkt bei
STARSOFT
Peinerstr. 81 D-3000 Hannover 81
Tel. 0511/837 99 77

Das seit über 3 Jahren bewährte Kopierprogramm copySTar gibt es jetzt in einer völlig neu überarbeiteten und optimierten Version.

Vollautomatische Erstellung von erlaubten Sicherheitskopien. Ihrer **Original-Software.** Raubkopien sind strafbar! Es sind **keine Parameter** Eingaben **nötig.** Kopiert **alle** Atari-St Disketten, ohne zusätzliche teure Hardware.

Superschnelle Kopien von "normalen" Disketten unter 30 Sekunden. Das Format der Disketten (9,10,11 Sektoren/SS-DS) wird dabei automatisch erkannt und richtig kopiert.

Automatische Fehlerkorrektur während des Kopiervorganges. Die Kopien von teilweise **defekten Disketten sind** oft **wieder** vollständig lauffähig und **fehlerfrei**.

Eine **umfangreiche Statusanzeige** beim Kopieren und Formatieren zeigt Ihnen genau, auf welchem Track/Sektor es Probleme oder Fehler beim Schreiben oder Lesen von Daten gegeben hat.

Superschnelle Formatierung von Disketten (9,10,11 Sektoren) Erweiterung der Disk-Kapazität auf über 912KB.

Virusprogramme werden erkannt und können gelöscht werden.

Konvertiert in **Spezialformat für doppelte Geschwindigkeit** ohne zusätzliche Hardware.

Qualitäts-**Prüfung von Leerdisketten**.

Drehzahl-Test des Disk-Laufwerkes. (Eine falsche Drehzahl kann die Ursache für viele Schreib-/Lesefehler sein.)

Super-Update-Service! Gem-Bedienung. **Der Preis DM 169,-**

Besitzer eines älteren copySTar erhalten die neue Version gegen eine geringe Gebühr. Bitte rufen Sie uns an.

► *PR's Disk-Speeder*

Das Harddisk-Beschleunigungsprogramm für den Atari ST. Durch ein intelligent optimiertes Cache-Prinzip erreichen Programme mit vielen Harddisk-Zugriffen **unglaubliche Geschwindigkeiten.**

Einfach und blitzschnell installierbar. Das Installations-Programm **analysiert die Harddisk** und errechnet die benötigte Speichergröße **automatisch**.

Frei konfigurierbar (Device·s, Speicherbelegung etc.)

Für den professionellen Anwender unentbehrlich. **DM 89,-**

Wichtig!! Alle Programme werden mit einem ausführlichem **deutschen Handbuch** und **ohne Kopierschutz** geliefert.

Alle Preise sind unverbindlich emfohlene Verkaufspreise



ST-FIBU

Die einfach zu bedienende Finanzbuchhaltung
Professionell – Schnell – Bedienungsfreundlich

Die einfach zu bedienende Finanzbuchhaltung
Professionell – Schnell – Bedienungsfreundlich

- Dialog-orientertes Buchen
- Konten anlegen ganz einfach beim Buchen
- Konten auch mit Namen suchen
- Anzeige vom Monatsjournal am Bildschirm
- Durchsuchen des Monatsjournals
- Berichtigen von Buchungsfehlern im Monatsjournal
- Aktueller Saldo beim Kontoaufruf auf dem Bildschirm
- Eingebauter Taschenrechner
- Druck aller Listen (Journal, Salden, Kontenplan etc.)
- Bilanz, Gewinn und Verlustrechnung
- Kontenblätter A5 oder A4
- Umsatzsteuervoranmeldung
- Offene-Posten Buchhaltung eingebaut
- Offene-Posten-Liste beim beim Buchen einsehbar
- Kein Kopierschutz, auch Festplattengeeignet!
- mit Handbuch
- Lauffähig auf jedem ST ab 512 KB u. SW-Monitor (SM 1124)

Vers. 1.04 für max. 52 Buchungen/Monat nur **DM 98,–**
Vers. 1.54 für max. 2400 Buchungen/Monat nur **DM 298,–**
Vers. 1.54 mandadntenfähig nur **DM 398,–**
Vers. 2.04 zusätzlich mit Mahnwesen, Textverarbeitung, Serienbrief, Formular nur **DM 498,–**
Vers. 2.04 mandantenfähig nur **DM 649,–**

Gestalten Sie sich jetzt Ihre Finanzbuchhaltung so, wie Sie es brauchen.

ST-Fibu 4.04

Finanzbuchhaltung wie Vers. 2.04 aber komplett in

GFA-Basic

Der GFA-Code ist in Prozeduren aufgebaut. Aus diesem Grund kann das Programm von jedem, der sich mit dieser Programmiersprache auskennt, seinen Erfordernissen angepaßt werden. Auch können z. B. beliebige Auswertungsprogramme erstellt werden. Die Möglichkeiten sind bekanntermaßen unbegrenzt.

Einzige Bedingung: Das Programm darf nur für den Bedarf des Käufers verändert werden und auch in verändertem Art nicht weiter verkauft werden.

Preis nur DM 1.200,–

Funktionsfähige **Demo**
(incl. Anleitung, wird bei Bestellung angerechnet) .... **DM 60,-**

Alle unsere Fibu Versionen sind untereinander Datenkompatibel!!!

Wenn Sie eine kleine Fibu Version besitzen, wird diese beim Kauf einer größeren Version angerechnet.



GEORG STARCK
Herzbergstraße 8 · D-6369 Niederdorfelden
☎ 06101/3007 –

Lieferung per NN + DM 8,– Versandkosten
b. Vork./V-Scheck Versandkostenfrei

BÜCHER

Titel	Autor	Verlag	Preis	Heft
Das TOS Listing	Kramer, Riebl Hübner	Heise Verlag	68,-	4/88
ATARI ST Profibuch	Jankowski, Reschke Rabich	Sybex Verlag	69,-	3/88
GEM-Programmierung in C	Aumiller, Luda Möllmann	M & T	69,-	11/88
C im Überblick	Burkhard	Heise Verlag	14,80	12/87
ATARI_ST Assembler-Buch	Wollschläger	M & T	59,-	10/87
Prog. in Maschinensprache	Nieber	Sybex Verlag	48,-	11/88
Dr. Dobb's Toolbox of 68000 Programming	Verschiedene	Prentice-Hall Press Verlag	64,-	6/88
GFA-BASIC Referenzhandbuch	Kofler	Sybex Verlag	49,-	11/87
GFA BASIC 3.0	Engels, Görens	GFA		7/88
Akt. Musterprogramme in BASIC	Lutz, Schulz-Osterloh	Interest	92,-	2/88
Programmieren mit FORTH	Aumiller, Luda	M & T	69,-	11/87
GEM Programmierung in Modula-2	Vogel	AXIS Verlag	25,-	5/88
Programmierpraxis ST Pascal	Wollschläger	M & T	59,-	10/88

BÜCHER

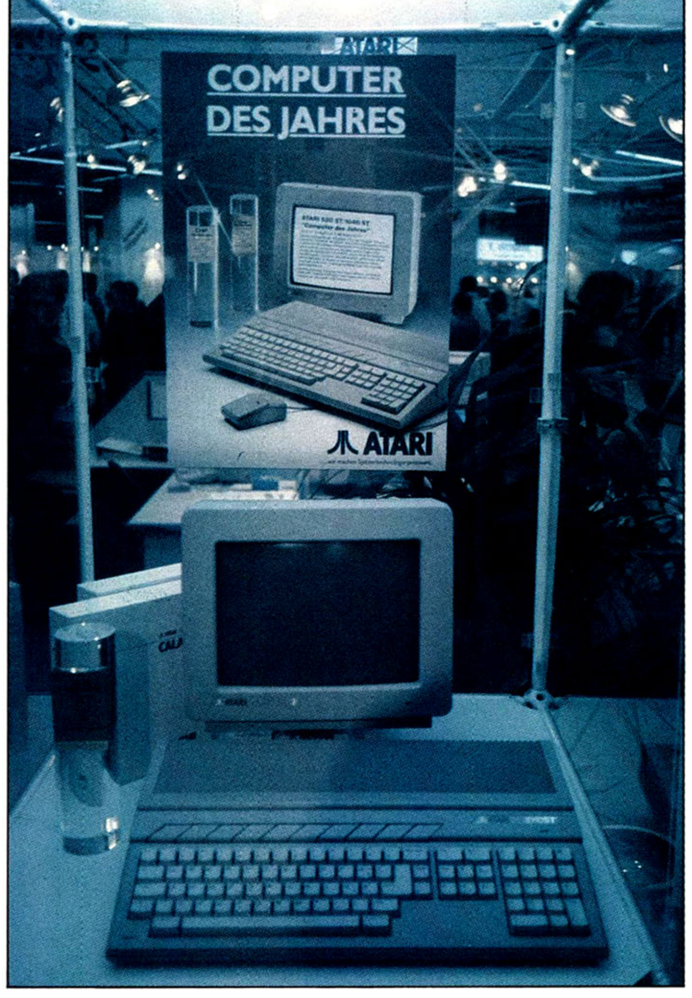
Titel	Autor	Verlag	Preis	Heft
Sortierverfahren in Pascal	Batz, Huschke	Franzis	28,-	4/88
LISP auf PCs	Haugg, Omlor	Hanser	48,-	9/87
PROLOG	Bratko	Addison Wesley		5/88
Occam Das Handbuch	Pountain, Rudolph	Heise	34,80	3/88
The Transputer Reference Manual	Verschiedene	Prentice-Hall Press Verlag	67,-	10/88
MIDI & Sound Buch	Dr.Enders, Klemme	M & T	69,-	8-9/88
Programmierung von Grafik und Sound	Mathy	M & T	52,-	12/87
Kompaktführer LaTex	Wonneberger	Addison Wesley	26,80	6/88
Signum!	Ritzhaupt	Application Systems	59,-	8-9/88
Adimens Praxis-Buch	Mollenhauer	M & T		7/88
Einführung in CAD	Liesert, Linden	DATA Becker	69,-	9/87
Scheibenkleister Massenspeicher am ST	Claus Brod Anton Stepper	MAXON	59,-	

Die Angaben zu den Software-, Hardware-Produkten und Büchern sind ohne Gewähr!

ORGATECHNIK KÖLN '88

Vom 20. bis 25. Oktober fand die 7. internationale Büromesse in Köln statt. Schwerpunkt dieser Fachmesse sind Büroeinrichtungen im weitesten Sinne und die Kommunikationstechnik. Für uns war natürlich die Anwesenheit von ATARI ein wichtiger Grund, die Orgatechnik zu besuchen.

Schon der Schwerpunkt dieser Fachmesse als auch die Ausrichtung auf Händler und berufliche Anwender und die damit verbundenen Eintrittspreise (Tageskarte DM 30,-, Dauerkarte DM 60,-) ließen den gewohnten "Messerummel" am ATARI-Stand vermissen. Dementsprechend waren dort auch keine sensationellen Neuvorstellungen zu entdecken.



Computer des Jahres 1988

Wieder einmal ist der ATARI 520 / 1040 STF zum Computer des Jahres gewählt worden. Fachjournalisten des Microcomputermagazins CHIP hatten den ATARI ST schon 1985 zum Computer des Jahres gewählt. Damals allerdings in der Rubrik "Personal Computer". Warum er dieses Jahr Sieger in der "Home Computer"-Klasse wurde, ist mir nicht ganz einleuchtend, wahrscheinlich ist der stark gesunkene Preis maßgeblich für diesen Klassensturz. Ob eine derartige Klassifizierung überhaupt sinnvoll und möglich ist, möchte ich an dieser Stelle offen lassen, jedenfalls ist der ATARI ST Sieger!

Megafile 20, 30 und 60 - neue Festplatten von ATARI

An einem der 16 Arbeitstische war sie zu sehen, die neue Festplattenserie *"Megafile"*. Schon lange bieten diverse Fremdhersteller Festplatten mit größeren Speicherkapazitäten als 20 Megabyte für den ST an. Doch nun hat auch ATARI zwei neue Platten mit 30 und 60 Megabyte im Programm. Die bisher verkaufte 20 MB-Platte SH 205 wurde umbenannt in Megafile 20. Sie wird wohl in Anbetracht des geringen Preisunterschiedes von nur DM 100,- zur Megafile 30-Platte kaum noch eine Daseinsberechtigung haben. Die neuen Platten sehen optisch genau wie die SH 205 aus und passen somit zur Mega ST-Serie. Sie unterscheiden sich nur in den Speicherkapazitäten und in den Preisen, die wie folgt lauten: Megafile 30 DM 1398,- und Megafile 60 DM 1998,-.

In Sachen Software wurde das neue TOS 1.4 als deutsche Diskettenversion gezeigt. Mit der Auslieferung der ROMs soll noch in diesem Jahr begonnen werden.

Platte, Streamer und Wechselplatte aus einer Hand

Die Firma Vortex zeigte in aufeinander abgestimmten Gehäusen ihre schon bewährte 60 MB-Festplatte zusammen mit einem Bandstreamer zur Datensicherung und einer 44 MB-Wechselplatte inkl. neuer Treibersoftware.

Man sollte beachten, daß die Orgatechnik eine allgemeine Büromesse ist und kurz zuvor in Düsseldorf eine ATARI-Messe stattfand. Alles in allem war die Orgatechnik für den reinen ATARI ST-Interessenten keine Show der Superlative.

UB

ALMO Statistik-System

Ein bewährtes Großrechner-Programm – jetzt auf dem ST
Ein Programm aus der Reihe der 'großen' Statistiksysteme

1. Häufigkeitsverteilung (mit vielen statistischen Maßzahlen)
2. Zwei- und beliebig-dimensionale Tabellierung. Mit Chi-Quadrat, Tschuprovs T, Kontingenzkoeffizient C, Lambda, Gamma, tau, r, punktbiseriales r, rho, Phi, Eta, t-Test, Wilcoxon-Test, Zeichentest, Ridits
3. Zellenmittelwerte: Mittelwerte und Streuungen quantitativer Variable je Merkmalskombination nominaler Variabler. Test auf Varianzhomogenität
4. Korrelationsmatrix, Kovarianz- und Quadratsummen-Matrix. Alle auch als partielle Matrizen. Einbeziehung ordinaler sowie nominaler Variabler
5. Allgemeines lineares Modell
   a. Regressionsanalyse b. Varianzanalyse c. Kovarianzanalyse
   d. Diskriminanzanalyse e. Logitanalyse f. Hierarchische Analysen
   g. Meßwiederholungs-Designs h. Analyse politischer Wahlen
   - Alle Analysen auch als multivariate mit mehreren abhäng. Variablen
   - Ordinale Variable. Nominale Variable als abhängige Variable möglich
6. Pfadanalyse: Regressionsanalyse rekursiver Kausalmodelle
7. Faktorenanalyse mit schiefwinkliger Rotation. Faktorwertberechnung
8. Clusteranalyse (7 Verfahren)
9. Rasch-Skalierungsverfahren und probabilistisches Unfolding
10. Latent Structure Analysis nach Lazarsfeld (alle Modelle)
11. Ähnlichkeits-Skalierung:
    Paarvergleich, Rangordnung, Tetradenvergleich, Tripelvergleich, multiple Rangordnung, Rangordnung von Paaren, Image-o. Profil-Analyse
12. Nichtmetrische MDS nach Kruskal: Analyse von Ähnlichkeitsmatrizen
13. Wählerstrom-Analyse. Allgemein: Quadratische Optimierung
14. Matrixoperationen: Inverse, Determinante, lineare Abhängigkeit

- Sorgfältige Behandlung fehlender Messwerte in allen Verfahren
- Integrierte Programmiersprache – alle denkbaren Variablen-Umkodierungen
- Relationale Datenbank-Operationen: Zugriff auf einzelne Datensätze. Bilden von Sub-Dateien. Hierarchische Dateistrukturen. Zusammenfügen von Dateien über Verbindungsvariable. Sortieren. Suchen über mehrere Dateien
- programmiert von Fachleuten von mehreren Universitäten
- GEM-Bedienung, Eingabe-Masken für alle Verfahren, integrierter Editor
- Handbuch (deutsch) über 600 Seiten, Mindestens 1 MB, 2-seitige Floppy

DM 198,- (+ Versand)
Demodiskette mit vollständigem ALMO limitiert auf 20 Variable und 60 Datensätze DM 20,-

Prof. Dr. Kurt Holm
Am Schlößlberg 8
A-4060 Leonding
Tel. 00 43-732-5 26 18

Hendrik Haase Computersysteme
präsentiert:

Atari-Computer

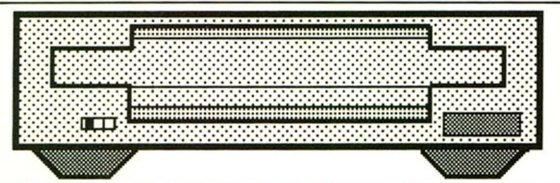
1040 STF incl. SM124	**1499,– DM**
Atari Mega ST2 incl. SM124	**2598,– DM**
Atari Mega ST4 incl. SM 124	**3598,– DM**
Drucker Star LC 10	**598,– DM**
Drucker NEC P6 plus	**1498,– DM**
NEC Multisync II	**1398,– DM**
Graustufen Multisync	**549,– DM**
Vortex-Festplatten HD20 plus	**1100,– DM**
Mitsubishi Multisync	**1298,– DM**

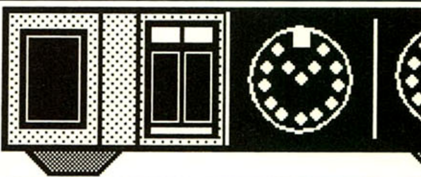
Bestellungen und Informationen bei:

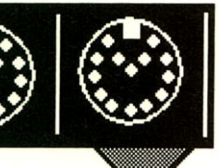
Hendrik Haase Computersysteme
Wiedfeldtstraße 77 · D-4300 Essen 1
Telefon: 02 01-42 25 75 · Fax: 02 01-41 04 21

PROFESSIONAL DISC DRIVES

FLOPPY-LAUFWERKE LUXUSAUSFÜHRUNG







PDD-SERIE

DIES IST PDD-3

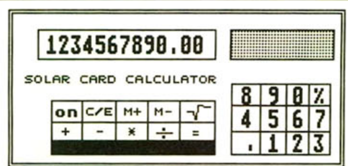
Grundausstattung: 100% ST-Compatibel

- ★ MIT NEC 1037a/1036a grau
- ★ MIT FLOPPYKABEL + NETZKABEL (> 1 m)
- ★ INTEGR. SPEZIALNETZTEIL + TRAFO (INTERN)
- ★ MIT STECKBAREN ANSCHLÜSSEN
- ★ 2 BUCHSEN, AN/AUS-SCHALTER
- ★ 720 KB – 950 KB FORMATIERT
- ★ 2. SCHALTER FÜR A/B VERTAUSCH

PDD-3	EINZELSTATION	325,–
PDD-13	DOPPELSTATION (2 × 720 KB)	565,–
PDD-5	5 1/4 Zoll, Umschalter vorne! 40/80 Tracks + Software, IBM-Compatibel	379,–
PDD-16	3 1/2 Zoll plus 5 1/4 Zoll in einem Gehäuse, wie PDD-3 plus PDD 5 zusammen	689,–
NEC 1037 A grau		169,–

WEITERE FLOPPYSTATIONEN (EINFACHE AUSFÜHRUNG):

ST-3	ST-13	ST-5	ST-16	NEC 1036 A
EINZELSTATION NEC 1037 A/1036 A KOMPL. ANSCHLUSSF.	DOPPELSTATION INTEGRIERTES NETZT. 2 × 720 KB	5 1/4 ZOLL, TEAK 55FR 40/80 TRACK	3 1/2 + 5 1/4 ZOLL NETZTEIL INTEGRIERT DOPPELSTATION	
239,–	488,–	329,–	625,–	159,– grau



JEDER BESTELLUNG LIEGT EIN SCHECKKARTEN-RECHNER BEI!

DIGITALE SPRACH-AUSGABE



DM 89,–
Der ST kann sprechen
Features:
Auf wissensch. Grundlage Sprachausgabe-Prg. voll in GEM
GFA-Quellcode und Handbuch
Deutsche Sprachverarbeitung!
Sehr guter Vokalgenerator
Perfekte A,E,I,O,U-Laute
Sehr gute Sch-ZZ-FFST u. a Konsonanten,
Silbenpausen, Speed
und vieles mehr einstellbar

Fischer Computer · Goethestr. 7 · 6101 Fr.-Crumbach · 0 61 64 / 46 01 abends zw. 20+22 Uhr ebenso



STARKE SOFTWARE

ST-KREATIV-DESIGNER

DAS SUPER-PROGRAMM ZUM KREATIVEN ERSTELLEN VON GRAFIKEN ALLER ART – OHNE MALKENNTNISSE

Das Selbstverständliche

Einfache Bedienung. Ansprechende Menüs. Alle Bildoperationen lassen sich rückgängig machen. Automatischer Malmodus, der Muster in Millionen Variationen erstellt.

- 2 Füllroutinen
- 2 Blockspeicher
- Absolut flimmerfreie Block- und Bildverarbeitung. Die eingestellte Verknüpfung ist schon beim Zeichnen und bewegen von Bildteilen zu sehen
- WYSIWYG! Blocks lassen sich in jeder Richtung über den Bildrand hinausschieben. Ausschneiden und Einkopieren von beliebigen Formen!
- Freihand: verschiedene Pinsel. Füllmuster als Pinsel, Malen mit Bildausschnitten. Spiegeln beim Zeichnen um 1 oder 2 Achsen
- Füllmuster definieren ohne Editor – eine ganz neue einfache aber wirkungsvolle Methode
- Vergrößern (Bild oder Text)
- Verkleinern, 3 Endprodukte zur Auswahl
- Spiegeln
- Drehen: dreht in 2 Richtungen gleichzeitig
- Stauchen
- Rahmen(1): Erzeugen von Rahmen aus allen Graphiken
- Rahmen(2): automatische Rahmenberechnung
- Schatten: automatische Schattenberechnung (3D Effekt)
- Teilmuster: aus Bildteilen (und Teilchen) können weitere Muster, auch Füllmuster erstellt werden
- Font's: 23 Größen, 21 Arten, 4 Verknüpfungen, weitere Verarbeitung durch Schatten. Rahmen usw. möglich. Schreibrichtung und Farbe kann während des Schreibens geändert werden.
- Lineal: Einblendbare Einteilung
- Radierer und Sprühdose in jeder Größe einstellbar. Zwei Sprühdosen Modi
- Schnelle Lupe mit: Punkt, Rechteck, Linie, Inventieren, Löschen
- Weitere Optionen: z. B. Bewegen, Kopieren, Kopieren nach..., Rechteck, Kreis, Linie, 3D-Rechteck, Preistafel...
- Invertieren auf Knopfdruck während gezeichnet wird
- Voll Mausgesteuert
- Kreativ-Designer – Bilder können in Signum 2 verarbeitet werden!
- Für Designer, Werbestudios, Druckgewerbe, Graphiker, aber auch für den nicht professionellen Anwender z. B. zum Erstellen einer Schülerzeitung, Handzettel, Grußkarten usw.

Das Besondere

- Fragen Sie mal die Bedienungsanleitung Ihres Zeichenprogramms ob folgende Funktionen möglich sind:
- Selbständige Hintergrund- und Füllmustererzeugung
- Über 200 Selbstdefinierbare Füllmuster im Speicher
- Kein Flimmern kein Gummiband
- Beim Zeichnen und Bewegen ist immer das fertige Endprodukt zu sehen
- Einige selbständige Rahmen und Schattenberechnungen
- Drei Vier Fünf... Neunundvierzigecke
- Automatisch Blocks ausschneiden
- Block paßgenau Einkopieren mit oder ohne Rahmen
- Blocks lassen sich in jede Richtung über den Bildrand hinausschieben
- Blaupause, Abmalen, Durchmalen von anderen Bildern
- Ein echtes Zeichenbrett mit Linealen
- Bilder maskieren
- Blocksatz, Proportionalsatz, Zentrieren

DM 128,–

* alle Preise sind unverbindlich empfohlene Verkaufspreise

ST-C.A.R.

Computer Aided Regulation

Das Programm zur Systemanalyse und Regelungssynthese

Zielgruppe / Aufgabenbereich

Studenten der Ingenieurwissenschaftlichen Fachrichtung

- Veranschaulichung der Auswirkung von Regelkreisglieder/-strecken
- Überprüfung erarbeiteter Lösungen (numerisch und grafisch)
- Einsatz in der Entwicklung eigener Regelsysteme (z. B. Diplomarbeit)

Ingenieure der technischen Fachrichtung

- Einsatz in der Entwicklung benötigter Regelkreise (Zeitersparnis)
- Systemanalyse vorhandener Regelungen (Untersuchungen möglicher Änderungen)
- Unterstützung beim Entwurf durch 'weniger Erfahrene' (Auffrischung des Wissens, eigenes Wissen überprüfen)

Institute, Lehr- und Forschungseinrichtungen des technisch-wissenschaftlichen Bereiches

- Einsatz in der Entwicklung benötigter Regelsysteme
- Analyse vorhandener Regelsysteme (Untersuchung möglicher Änderungen)
- Hilfe bei der Erstellung von Lehr-(Unterrichts-)Material
- Direkter Einsatz im Unterricht ('am lebenden Objekt')

C.A.R. ...das Programm mit umfassender Leistung
C.A.R. ...ausführliche Anleitung mit Beispielen
C.A.R. ...das Programm für professionelle Ansprüche

DM 198,–

ST DISK BOX

DISKETTEN-VERWALTUNGSPROGRAMM

- Ablegen der Disketten nach Nummern (650 Stück)
- Ablegen in freibenennbaren Sparten
- Mitablegen von Texten möglich
- Suchen nach Programmname, Disknummer, Text, Extender
- Löschen durch Mausklick
- Druckfunktion für Disknummer, Extender, Sparte
- Programm erkennt doppelte Einsortierung in eine Datei
- Programm ist voll GEM-unterstützt
- große Geschwindigkeit
- komfortabel zu bedienen
- Nachträgliches Bearbeiten der Dateien möglich

DM 49,–

SALIX-PROLOG

leistungsfähiges KI-Programmiersystem

- Edinburgh-Standard
- schnell, ca. 1200 LIpS
- ca. 160 eingebaute Funktionen
- Gleitkommaarithmetik
- läuft unter GEM
- ca. 140 GEM Library Funktionen
- Datenbankeditor
- Einbindung von Fremdeditoren
- zyklische Strukturen werden verarbeitet
- Exception Handling
- leistungsfähiges Testsystem

Jetzt neu: Version 2.6

- benutzerdefinierte Funktionen
- globale Variable
- Clipboard-Device

DM 198,–

Update von 2.0 – 2.5 DM 49,–

SALIX DIALOGEXPERTE

Wer schon einmal eine Anwendung unter GEM geschrieben hat – mit Drop-down-Menüs, Dialogboxen und Alerts, wird es wissen:

So einfach sich die einzelnen Objekte mit dem Resource Construction Set Editor erzeugen lassen, so umständlich ist ihre Einbettung in das Anwendungsprogramm.

Es geht auch anders: SALIX hat jetzt einen DIALOG-Experten entwickelt, der die RSC- und die DEF-Dateien selbständig untersucht. Dann geht's erst richtig los: **„Programming by doing"** heißt die Device. Zunächst werden die Fenster der Anwendung definiert – mit der Muas. Dann wird die Bedeutung der Menüpunkte festgelegt, einfach indem man sie anklickt und das aufzurufende Prädikat eingibt.

Mehrere Menüleisten sind zulässig. Die einzelnen Menüpunkte lassen sich in logische Gruppen zusammenfassen, das Check-Symbol wird kontrolliert, und und und...

Die Knöpfe der Dialogboxen und Alerts lassen sich symbolisch ansprechen, eine Reihe von Prädikaten ermöglicht den einfachen Zugriff auf die GEM-Objekte von der Anwendung aus, komplexe Formulare können auch dynamisch aus der Anwendung heraus aufgebaut werden.

Sie haben die Wahl: sich weiterhin damit abzuquälen, Menüpunkte auszuzählen, oder sich lieber auf das Design und Logik Ihrer Anwendung zu konzentrieren.

Der Dialogexperte läuft unter SALIX PROLOG 2.6.

DM 79,–

ST PRINT Neue Version

DAS VIELFACH BEWÄHRTE MULTIACCESSORY JETZT IN ERWEITERTER VERSION V2.1 INCL. VIRUSKILLER

ST-PRINT erleichtert durch seine Vielseitigkeit und einfache Bedienung die tägliche Arbeit. Insbesondere Anpassungs- und Verständigungsprobleme zwischen Computer und Drucker werden gelöst.

Parameter -load und -save alle Einstellungen (Ramdisk, Spooler, Hardcopy und Druckervoreinstellung) werden abgespeichert und können auch beim Booten automatisch eingestellt werden.

1. Resetfeste Ramdisk

- Größe einstellbar von 32 – 4000 KB, auf Wunsch resetfest oder abschaltbar
- kann auf Laufwerk C bis P gelegt werden
- arbeitet problemlos mit einer Harddisk zusammen

2. Druckerspooler

- Größe einstellbare von 2 – 510 KB
- abschaltbar
- arbeitet mit TOS- und GEM-Programmen
- auch für Hardcopies
- in Maschinensprache programmiert und interruptgesteuert, dadurch immer optimale Geschwindigkeit beim Ausdruck
- Größe nun 2 – 4000 kByte

3. Druckervoreinstellung

- komfortable Druckereinstellung mit der Maus: Knopfdruck statt Handbuch oder DIP-Schalter
- viele Einstellmöglichkeiten: Zeilenvorschub, Schriftart, Zeichensatz, Papierränder etc.
- Einstellung des Druckers vom Desktop und aus jedem GEM-Programm (VIP-Professional, Wordplus, Tempus etc.) möglich
- **kann an alle Matrixdrucker angepaßt werden, diverse Druckertreiber gehören zum Lieferumfang, auch eigene Anpassungen können erstellt werden**
- **Druckeranpassung jederzeit nachladbar**

4. Druckerzeichenkonverter

- Anpassung aller Atari-Zeichen an den verwendeten Drucker, ermöglicht den korrekten Ausdruck von Umlauten und Sonderzeichen auf allen Matrixdruckern
- Umlaute und Sonderzeichen können gemischt verwendet werden
- endlich problemloser Ausdruck mit Programmen (TOS oder GEM), die keine eigene Druckeranpassung haben

5. Hardcopyroutine

- nutzt die Fähigkeiten von 9-, 18- und 24-Nadeldruckern
- kann an jeden Matrixdrucker angepaßt werden
- verschiedene Auflösungen, Schnelldruck bis Qualitätsdruck
- arbeitet mit dem eingebauten Spooler zusammen *gespoolte Hardcopy
- Auslösung der Hardcopy durch Alternate/Help, kann genauso wieder abgebrochen werden
- Umsetzung der Farben in Grauwerte (im MID-RES Modus)
- nun auch Ansteuerung HP-Laserjet-kompatibler Laserdrucker
- Hardcopy als Bild auf Diskette, Ramdisk oder Harddisk im Degas- oder Screen-format

6. Viruskiller

- Erkennen und Vernichten von BOOT-SEKTOR-Viren
- GEM-Programm, durch einfaches Umbenennen (.ACC statt .PRG) auch als Accessory verwendbar

DM 69,–

Update DM 19,–

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir: ____________________

____

zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Auslieferung in der Schweiz:

DataTrade AG
Langstr. 31
CH-8021 Zürich

Keep Hacking

Fremdtastaturen am ST

Sollten Sie die Werbung in dieser Zeitschrift ausführlich durchlesen, werden Sie feststellen, daß außer einer Originalplatine (und die gibt es auch schon z.B. im 19"-Gehäuse) eigentlich alle Peripheriegeräte von Fremdanbietern geliefert werden. Diese Produkte, Monitore, Laufwerke, Tastaturen usw. sind, um konkurrenzfähig zu sein, entweder viel billiger oder ganz einfach besser. Um den zwischenzeitlich großen Markt an Fremdtastaturen etwas zu durchleuchten, wollen wir Ihnen unsere Erfahrung mit vier verschiedenen Lösungen vermitteln, die alle ihre Vor- und Nachteile haben.

Direkter Anschluß

Anschlußmöglichkeiten boten sich über die Midischnittstelle, den ROM- Port und das Einschleifen direkt in den Tastaturanschluß. Wem diese Anschlüsse nicht zusagen, kann auch die 100% kompatible, die alte Tastatur ersetzende *MTST*-Tastatur von Binnewies anschließen. Die letztgenannte Möglichkeit ist natürlich die perfekte, aber nicht ganz billige Lösung der Tastaturfrage. Dafür wird die Verträglichkeit mit allen Programmen sowie allen Emulatoren garantiert.

Für die Arbeit mit Emulatoren benötigt man bei den Tastaturen, die an eine Schnittstelle angeschlossen werden, entweder ein anderes Treiberprogramm, oder der Emulator selbst muß gepatcht werden. Die Beschreibung eines PC ditto-Patchs wird der Midi-Lösung beigelegt, an Änderungen von Aladin und ... arbeitet die Vertriebsfirma. Die MTST-Tastatur hat selbstverständlich ihre eigene Maus- und Joystickbuchse und als Zugabe einen Hardwarereset, der über zwei zusätzliche Tasten ausgelöst wird. Dies sind auch die einzigen neu hinzugekommenen. Die anderen Tasten sind mit einer Ausnahme auf dem von der alten Tastatur her gewohnten Platz geblieben. Der Reset ist allerdings an einem MEGA ST nicht möglich, dafür gestaltet sich der Anschluß an diesen besonders einfach: Stecker der alten Tastatur ziehen, neuen hineinstecken und fertig (die alte MEGA-Tastatur läßt sich bestimmt gut verkaufen). Sollten Sie nicht glücklicher Besitzer eines MEGA STs sein, gestaltet sich der Umbau etwas komplizierter. Sie müssen den Rechner aufschrauben, die Steckverbindung der Tastatur lösen und die neue anstecken (Garantieverlust). Da zum Test nur die Tastatur für den MEGA ST zur Verfügung stand, kann ich nicht beurteilen, ob sich dabei Schwierigkeiten ergeben.

Das Schreibgefühl der MTST-Tastatur ist allerdings so gut, daß man sich von solchen Kleinigkeiten nicht abhalten lassen sollte. Die von mir getestete Tastatur hatte einen Hub von 2.5 mm und einen fühlbaren Druckpunkt. Lieferbar ist sie auch ohne Druckpunkt bzw. mit 4 mm Hub, jeweils zu einem Preis von DM 589,-. Sollten Sie den Prozessor aus Ihrer alten Tastatur weiter benutzen wollen, kann ein Preisnachlaß von DM 35,- gewährt werden. Sollte auch auf das ergonomische Gehäuse mit Handballenauflage verzichtet werden können, verringern sich die Kosten noch einmal um DM 55,-. Der Preis mag hoch erscheinen, zumal die kleinen STs noch billiger als die Tastatur sind, überteuert ist die MTST-Tastatur aber auf keinen Fall. Qualität hat eben ihren Preis.

Sollte diese Anschaffung zu kostspielig sein, und sollten Sie auf vergoldete Kontakte in den Mikroschaltern verzichten können, kann ich die zweite von mir getestete Tastatur empfehlen. Es handelt sich dabei um eine *umgerüstete AT-Tastatur*, die von *H.Velder* umgebaut und vertrieben wird. Sie wird für DM 249,- anschlußfertig für den Midi-Port geliefert. Sollte dieser bei Ihnen schon belegt sein oder später belegt werden, kann der Anschluß auch parallel zur ST-Tastatur erfolgen. Das dafür notwendige Adapterkabel soll für zusätzliche DM 25,- zu beziehen sein. Ein Prototyp des Adapterkabels konnte an meinem 1040er problemlos zwischen Tastatur und Rechner geschaltet werden. Die ganze Prozedur hat mit Aufschrauben und erstem Test fünf Minuten gedauert. Der Umbau ist also auch für völlig ungeübte Bastler kein Problem.

Vorteil dieser Lösung ist die Unabhängigkeit von einem Treiberprogramm, wodurch das Keyboard auch mit jedem Emulator problemlos zusammenarbeitet. Das heißt nicht, daß die alte Tastatur völlig ersetzt wird, die Mausdaten und die Uhrzeit werden noch von ihr benötigt, und ganz problemlos geht es wegen dieser konfliktreichen parallelen Verbindung eben doch nicht. Der einzige mir bekannte Fehler trat bei schneller Eingabe von Zeichen und gleichzeitiger Mausbewegung auf. Die Tastatur ist zwar so geschaltet, daß bei einem kurzen Ruck an der Maus die Zeichen zurückgehalten werden, also die Mausdaten auch richtig beim Rechner ankommen; aber müssen bei ständiger Bewegung der Maus zuviele Zeichen zurückgehalten werden, geht es erst weiter, wenn eine Taste auf der alten Tastatur gedrückt wird. Solange stellt sich die AT-Tastatur tot bzw. sendet wahllos Daten. Immerhin, es geht danach noch weiter. Bei meinem ausgiebigen Test ist es mir allerdings erst passiert (gelungen), als ich in den Unterlagen von eben diesem Problem gelesen und die Fehlfunktion provoziert habe. Also kein Grund zur Panik. Der Anschluß ist sicher.

Mit Treibern

Werden keine Emulatoren verwendet, und ist die Midi-Schnittstelle unbenutzt, ist gegen den Anschluß an sie nichts einzuwenden. Das Treiberprogramm wird aus dem Autoordner gestartet oder U7 und U4 der ROMs geändert und schon funktioniert das Ganze. Die oben beschriebene Datenkollision wird durch den Treiber abgefangen.

Wie Sie im Bild sehen können, wurden AT-Tastaturen mit 102 Tasten getestet. Da der ST weniger hat, bleiben einige unbelegt sollte man zumindest annehmen. Dem ist aber nicht so. Help und Undo, die auf der AT-Tastatur fehlen, liegen hier auf F11 und F12. Die Tasten Druck, Rollen und Print sind mit besonderen Optionen belegt. Control (hier mit STRG bezeichnet) links, rechts und Druck gleichzeitig betätigt, schaltet die Tastatur z.B. in einen Vorwahlmodus. Hier kann ausgewählt werden, wie sich die "überflüssigen" Tasten verhalten sollen. Als Beispiel soll die Voreinstellung für Tempus genannt werden. ROLLEN bzw. Shift-ROLLEN bewirkt einen Sprung auf die nächste Bildschirmseite, die ENDE-Taste läßt den Cursor an die letzte Position springen, und Druck ruft die Suchfunktion auf. Sollten Sie eigene Vorstellungen einer Programmführung haben, wird sich die Vertriebsfirma sicher über Ihre Anregung freuen. Als Bonbon für Signum-Benutzer ist eine Control- und Alternate-Verriegelung einzuschalten. Dabei fühlen sich sinnvollerweise nur die Zeichentasten angesprochen. Zur Verdeutlichung: Aktivieren Sie diese Option, verhalten sich die Tasten bei Betätigung, als wäre die Control- bzw. Alternate-Taste auch gedrückt. Also sehr sinnvoll, wenn diese Ebenen mit einem anderen Zeichensatz belegt sind.

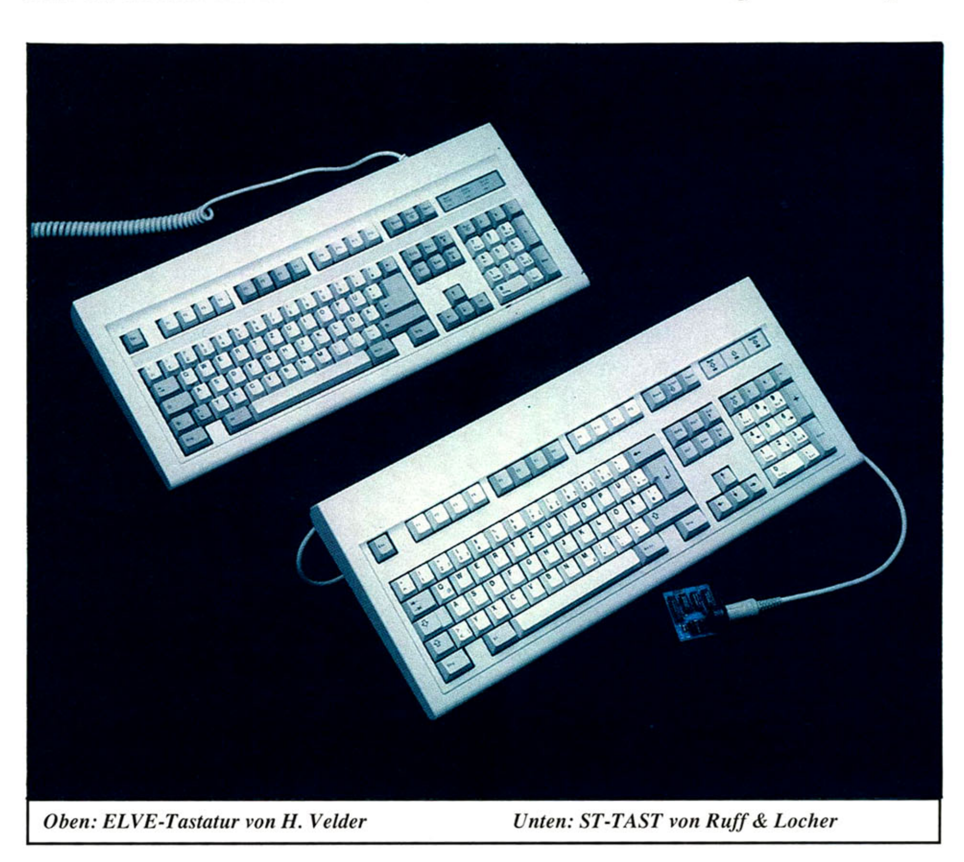
Ach ja, den genauen Anschluß an Midi wollte ich noch erklären. Der gestaltet sich so einfach, daß ich es fast vergessen hätte. Also, Stecker des Zusatzkabels in Midi-In, den Joystickstecker - na wohin wohl - in die Joystickbuchse und das Tastaturkabel in die noch freie fünfpolige Kupplung am Kabel der Tastatur stecken, schon kann es losgehen.

Zur technischen Ausführung und zum Eindruck gibt es wenig zu sagen. Die Tasten haben einen gekapselten (was immer das heißt) Mikroschalter mit deutlich hörbarem Tastaturklick und natürlich den dazugehörigen Druckpunkt. Das Gepiepse aus dem Monitor gehört der Vergangenheit an, wenigstens bei mir zu Hause. Das Schreibgefühl ist hervorragend (subjektiv), die ebenfalls von *Velder* gelieferte *Mitsumi*-Tastatur (die ca. DM 280,- kosten soll) mit nicht hörbarem Druckpunkt konnte mich auch nicht noch mehr begeistern. Damit ich nicht falsch verstanden werde, die beiden sind wirklich gut.

Die Mitsumi-Tastatur ist allerdings nicht umgerüstet. Sie benötigt eine Zusatzhardware, die in einer kleinen Box untergebracht ist und neben,unter ... den ST gestellt wird. Diese Box stand als Prototyp zur Verfügung, ist aber bei Erscheinen dieser Ausgabe bereits in Stückzahlen erhältlich. Der Anschluß erfolgt, wie oben bereits beschrieben, an den Midi-Port. Sie arbeitet zur Zeit nur mit dieser Schnittstelle zusammen, kann also nicht parallel zur Originaltastatur betrieben werden. Noch nicht, denn die Firma H.Velder versucht, die Leistungsdaten noch zu erhöhen. Schraubt man das Gehäuse auf, sieht man eine gut verarbeitete Platine (und das in der Testphase, alle Achtung), auf der zwei Steckplätze frei sind. Ich könnte mir denken, daß dort EPROMs oder ähnliches hinein sollen.

Da an dieser Box alle IBM-Kompatiblen Tastaturen anzuschließen sein sollen, diese sich aber wenigstens bei einem Reset oder beim Einschalten des Rechners an keinen erkennbaren Standard halten, ist die korrekte Verarbeitung der ankommenden Daten eine aufwendige Angelegenheit. Wie mir Herr Velder telefonisch mitteilte, ist es ihm nach erheblichen Schwierigkeiten gelungen, seine Box an alle ihm bekannten Tastaturen anzupassen. Sollten Sie also Ihre AT-Tastatur, die Sie schon zu Hause rumliegen haben, benutzen wollen, kann eine Nachfrage, ob sie mit der *ELVE KEY-Box* (so ist ihr voller Name) zusammenarbeitet, nicht schaden.

Neben der Tastatur von Mitsumi konnte nur eine *Cherry G 80-1000* an der Box getestet werden, wobei auch keine Probleme auftraten. Ansonsten sind die Leistungen noch nicht so gut wie bei der oben beschriebenen Lösung mit der umgerüsteten Tastatur (z.B. fehlt die Control-Verriegelung ..). Die von der ATARI-Norm abweichenden Belegungen wie der vertikale Strich auf *Alternate* + < sind optional abschaltbar, um die Übermittlung dieses Codes zu ermöglichen. Diese Tastenkombination löscht in Tempus z.B. bis zum Zeilenende. Wenn ich sage abschaltbar, dann heißt es nicht, daß dieses Zeichen nicht mehr erreichbar wäre, es liegt noch auf einer anderen Taste. Die Simulation der Maus über Tastatur ist möglich, d.h. die Cursortasten bewegen den Mauszeiger jeweils acht Pixel in die gewählte Richtung, die entsprechenden



Oben: ELVE-Tastatur von H. Velder *Unten: ST-TAST von Ruff & Locher*

GE – MEGA-DRIVE

★ Anschlussfertige Festplatten für die ATARI-Computer der ST-Serie

Formatierte Kapazität
★ 40,2MB MFM
zweites Laufwerk mit ST-506 Schnittstelle nachrüstbar (max. 512MB)
★ 61,5MB RLL
★ 80,5MB MFM
123,0MB RLL

Booten
★ durch Bootmenü freie Auswahl der Bootpartition von A: bis Z: möglich ★ Die Autostartprogramme und Accesories sind beim Booten frei wählbar.



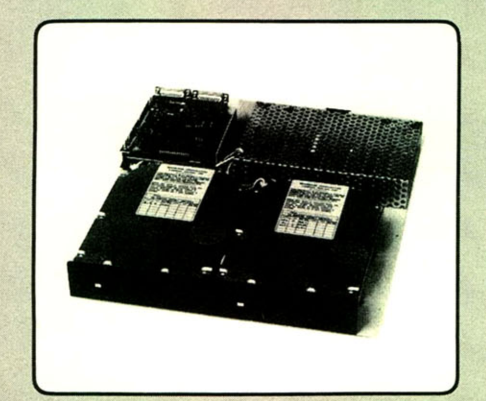
Treiber
★ AHDI-kompatibler Treiber, daher auch mit anderen Betriebsystemen lauffähig (z.B. PC Ditto, Aladin, RTOS)

Partition
★ in 16 Partitions mit 16/32MB (altes/neues IOS) einteilbar.

Software
★ Installationsprogramm, Treiber, Bootmenü, Backup

DMA-Anschluß
★ durchgeschleifter und gepufferter DMA-Ausgang zum Anschluß weiterer DMA-Geräte ★ Die DMA-Adresse ist von außen einstellbar.

Netzteil
★ leistungsstarkes Schaltnetzteil 150W



Gehäuse
★ massives Stahlblechgehäuse mit grauem Struckturlack und Netzschalter an der Frontblende

GE SOFT Computersysteme

Die Produkte der Firma GE-SOFT können Sie auch über Ihren ATARI-Fachhändler beziehen.
Distributor für Schweiz, Österreich und die Niederlande gesucht.

Maße
★ H x B x T 52 x 343 x 356mm

GE - MD 40	**DM 1598.-**
GE - MD 60	**DM 1798.-**
GE - MD 80	**DM 2398.-**
GE - MD 120	**DM 2598.-**

MEGA-DRIVE-Hostadapter
★ wie im MEGA-DRIVE eingebaut zum Betrieb einer Festplatte ist zusätzlich noch ein PC-Festplattencontroller OMTI 5520/5527 erforderlich ★ inclusive Software

GE - MDH	**DM 348.-**
mit OMTI 5520 B (MFM) **GE - MDHM**	**DM 448.-**
mit OMTI 5527 B (RLL) **GE - MDHR**	**DM 498.-**

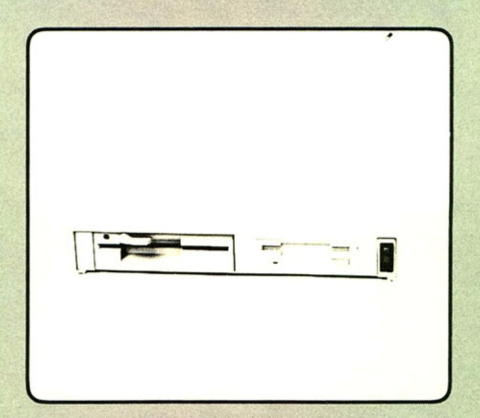
MEGA-DRIVE-Gehäuse
★ mit Schaltnetzteil, Hostadapter, OMTI und Software

mit OMTI 5520 B **GE - MDGM**	**DM 698.-**
mit OMTI 5527 B **GE - MDGR**	**DM 748.-**

GE – MEGA-CHANGE

★ RICOH 20MB Wechselplatte anschlußfertig für ATARI ST ★ technische Daten wie GE – MEGA-DRIVE

GE - MC 20 **DM 2498.-**



GE – MIX-DRIVE

★ Doppeldiskstation im MEGA-DRIVE-Gehäuse ★ NEC 1036A 3,5" und TEAC 5,25" ★ Schaltbar als Doppelstation oder umschaltbares B: Laufwerk ★ 5,25" Traccumschaltung 40/80 ★ eingebautes Schaltnetzteil

GE - XD 5,25"/3,5" **DM 898.-**

GE – DUO-DRIVE

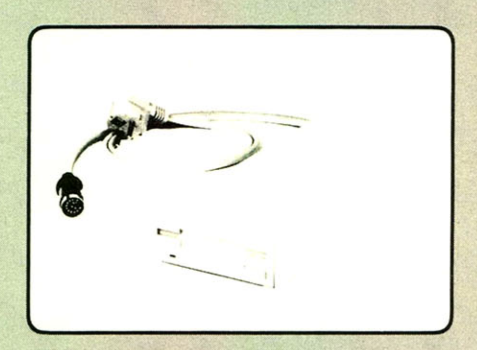
★ Doppeldiskstation ★ zwei NEC 1036A 3,5" im MEGA-DRIVE-Gehäuse ★ eingebautes Schaltnetzteil

GE - DD 3,5" **DM 898.-**

GE – MEGA-DRIVE-CHANGE

★ Festplatte 40MB und Wechselplatte 20MB

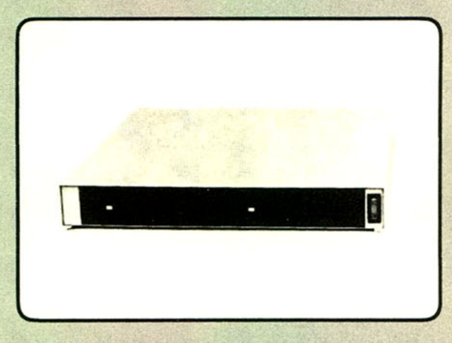
GE - MDC 40/20 **DM 3298.-**



GE – DISK-DRIVE

★ NEC 1036A Disklaufwerk im Stahlblechgehäuse mit integriertem Netzteil ★ durchgeschleifter Bus

GE - D 3.5" **DM 348.-**



GE – MEGA-MIX

★ Festplatte mit 40 oder 60MB mit Diskettenlaufwerk ★ NEC 1036A 3,5" oder TEAC 5,25" 40/80 Track

GE - MX 40/3,5"	**DM 1948.-**
GE - MX 60/3,5"	**DM 2148.-**
GE - MX 40/5,25"	**DM 1948.-**
GE - MX 60/5,25"	**DM 2148.-**

GE – MEGA-CHANGE-MIX

★ Wechselplatte 20MB mit Diskettenlaufwerk ★ NEC 1036A 3.5" oder 5.25" 40/80 Track

GE - MCX 20/3,5"	**DM 2848.-**
GE - MCX 20/5.25"	**DM 2848.-**

GE-SOFT
Computersysteme
Graurheindorferstr. 69
D-5300 Bonn 1

☎ **(02 28) 69 42 21**



Entwurf + Druck PLUM, Düren-Oberzier, TEL. 0 24 28/7 11

Tasten auf dem Zehnerblock bewirken die pixelweise Bewegung. Die beiden Maustasten sind über Enter bzw. das Pluszeichen erreichbar. Für Anwendungen, bei denen es auf sehr genaue Positionierung der Maus ankommt, ist es genau die passende Lösung.

Sollten Sie sich für eine Original-AT-Tastatur entscheiden, sollten Sie sicherstellen, daß sie wirklich besser ist als die des STs. Also unbedingt mal draufrumhämmern (stellt Sie natürlich bei Erwerb über einen Versand vor einige Probleme). Die oben erwähnte Cherry-Tastatur zum Beispiel konnte mich in keiner Weise begeistern. Schwabbelig wie die des ST (subjektive Meinung), zumindest nicht besser als die des MEGA-ST, die mir auch nicht zusagt.

Die letzte hier vorgestellte Lösung ist *ST-TAST* der Firma *Ruff & Locher*, die ebenfalls den Anschluß nicht umgerüsteter, IBM-kompatibler Tastaturen erlaubt. Der Anschluß erfolgt mittels einer sehr kleinen, offenen Platine, die in den ROM-Port gesteckt wird. Die beiden Originaltastaturen arbeiteten auch hier ohne Probleme, fast ohne. Der einzige Fehler, der mir auffiel, und das recht schmerzlich, war die falsche Belegung der Bindestrichtaste. Es ist zwar möglich, das Minuszeichen auf dem Zehnerblock zu benutzen, aber dem Schreibfluß bei meinen Zehnfingerschreibversuchen ist das nicht dienlich. Außerdem war die falsche Belegung nicht von Dauer, d.h. einmal lagen der Bindestrich und der Unterstrich richtig, das andere Mal, ohne Reset oder das Textprogramm verlassen zu haben, konnte mit dieser Taste nur noch der Schrägstrich erreicht werden (die Götter wissen warum). Ich kann nur wage vermuten, daß es mit dem ROM-Port zusammenhängt. Wie Sie vielleicht wissen, ist der ROM-Port nicht interruptfähig. Die an ihm angeschlossene Hardware kann also keinen Interrupt auslösen. Das Treiberprogramm muß sich in die VBL-Interruptliste einhängen und ständig den Port auf eingehende Daten abfragen, wobei es meiner Meinung nach zu Datenverlusten oder Verfälschungen kommen kann. Ich nehme allerdings an, das an einer Verbesserung des Treiberprogrammes, sofern es für den Fehler verantwortlich ist, gearbeitet wird. Ich persönlich boote lieber meinen Rechner weiterhin von der ROM-Disk, anstatt diesen Port mit einer Tastatur zu blockieren.

Positiv an dieser Lösung sind der gut erreichbare Anschluß ohne zusätzliches Kabel (wenn auch ohne Gehäuse) und die Möglichkeit, beim Laden des Treibers eine Tastaturtabelle mitzuladen. Auf der mitgelieferten Diskette sind neben der Standardbelegungstabelle noch drei weitere zu finden, die je nach Anwendung und eigenem Geschmack geladen werden können.

Fazit

Sollen alle Fehler ausgeschlossen werden, und wird auf eine qualitativ sehr hochwertige Tastatur Wert gelegt, sollten Sie sich für die MTST-Tastatur entscheiden.

Wollen Sie sich mit einer guten AT-Tastatur begnügen, wählen Sie am besten die umgerüstete der Firma Velder, die sehr zuverlässig arbeitet und dazu sehr preiswert ist. Diese verrichtet mittlerweiler in meinem alten 260er ihren Dienst. Ich habe sie direkt an die alte Tastatur angeschlossen.

Soll eine schon vorhandene Tastatur Anschluß finden, und sollte Sie die zusätzliche Box auf Ihrem Schreibtisch nicht stören, kann ich zu deren Anschaffung raten.

Die ST-TAST-Lösung der Firma Ruch & Locher hat zwar auch ihre Aufgabe erfüllt, doch sie kostet genauso viel wie die ELVE-Box, belegt aber den wichtigen ROM-Port.

Ein Problem möchte ich am Schluß doch noch erwähnen. Sollten Sie eine Bildschirmdunkelschaltung installiert haben, dürfte sich der Bildschirm mit Hilfe der externen Tastatur nicht hell schalten lassen. Zumindest war es bei meiner Dunkelschaltung erst nach Änderung des Dunkelschaltprogrammes möglich.

Nachträglich...

...ist nun auch die *AT-A-ST* Fertigversion eingetroffen. Sie mußte wegen eines Defektes wieder an die Firma *3K* zurückgeschickt werden. Den Fehler in der ersten Version konnte ich leider nicht lokalisieren, wohl aber den in der zweiten. Diese wollte zunächst auch nicht ihren Dienst aufnehmen, weßhalb ich doch stark an meinen Fähigkeiten, ein paar Stecker anschließen zu können, zweifelte.



Oben: ELVE-KEY-Box mit Mitsumi-Tastatur von H. Velder   Unten: MTST-Tastatur von Binnewies

Nach einigen Versuchen mit einem Ohmmeter konnte ich als Übeltäter ein kontaktloses Beinchen einer kleinen Zusatzplatine ermitteln und den Fehler beheben. Diese Platine sollte so behutsam wie nur möglich behandelt werden. Die Leiterbahnen sind teilweise nur so dick wie ein Haar und nicht gegen Beschädigung geschützt, die dickeren Leiterbahnen weisen mitunter einige Löcher auf. Das Ganze wirkt wenig professionell, und Reklamationen scheinen vorprogrammiert.

Der Anschluß von AT-A-ST erfolgt mittels dieser Platine. Dazu muß sie unter

dem Tastaturprozessor plaziert werden. Von dieser Platine, auf der sich nur ein 74LS07 befindet, wird ein Kabel nach außen zur externen Box geführt. Diese Box benötigt überflüssigerweise noch eine Box zur Stromversorgung. Es müssen also zwei zusätzliche Gehäuse und massig Kabel untergebracht werden. Wenn die Firma 3K unbedingt einen eigenen 8 Bit-Rechner (ja, so einer ist wirklich in der Schachtel) braucht um eine Tastatur am ST betreiben zu können, hätte man die Schaltung vielleicht besser in ein 19"-Gehäuse eingebaut, dann wüßte man wenigstens wofür knappe 300 Deutsche Mark bezahlt wurden (ohne Tastatur) und hätte nicht den Kabelsalat. Aber wer schaut schon unter den Schreibtisch. Zur Rechtfertigung muß wohl auf die Möglichkeit der Erweiterung hingewiesen werden. Das System läßt sich um einen Barcodeleser erweitern, der komplett allerdings 998,- DM kostet.

Sollte jetzt jemand glauben, nur das Teuerste sei gut genug für ihn, soll sich später nicht bei mir beschweren. Die Tastatur funktioniert zwar bestens am ST, parallel zur Originaltastatur, doch würde ich gerne wissen, weshalb es auf ihr nicht möglich ist in Tempus mit Shift+Insert eine Zeile einzufügen. Softwareprobleme im 8 Bit-Rechner? Sollte es sich um ein solches handeln, wird der Fehler (oder gibt es noch mehr?) demnächst behoben werden.

WS

Bezugsadressen:

MTST_Tastatur — *DM 589,-*

Binnewies datasystems
Bergfeldstr. 37
3000 Hannover 91
Tel. 0511/431006

ELVE-Tastatur (anschlußfertig) — *DM 249,-*
dazu Adapterkabel (direktanschluß) — *DM 25.-*
ELVE-KEY-Box — *DM 149,-*
Mitsumi AT-Tastatur — *DM 280,-*

H.Velder
Büro für Elektrotechnik
Provinzstr. 104
1000 Berlin
Tel. 030/4922754

ST-TAST (Platine+Software) — *DM 144,-*
Cherry G 80-1000 (AT-Tastatur) — *DM 248,-*
beides im Packet — *DM 342,-*

Ruff & Locher Datentechnik
Eichachstr. 13
7404 Ofterdingen
Tel. 07473/22810

AT-A-ST — *DM 298,-*

3K Kreidel-Knops-Kreidel
EDV Entwicklungen
Hülser Str. 76
4154 Tönisvorst

ENDE



PROFIPORT

Das Tor zur Welt

Im dritten Teil wollen wir versuchen, Ihnen die Programmierung des Profiports näher zu bringen. Port bedeutet eigentlich nur 'Tor', und genau darum dreht es sich hier - ein Tor zur (elektronischen) Außenwelt des Rechners.

In Abbildung 1 sehen Sie das Anschlußbild des Profiports. An dieser 64-poligen Messerleiste sind genau 32 individuell programmierbare digitale Leitungen *(1PA0-1PA7,1PB0-1PB7,2PA0-2PA7,2PB0-2PB7)*, 2 digitale Sonderleitungen *(1CB2,2CB2)* und 2 flankenempfindliche, digitale Eingänge *(1CB1, 2CB1)* verfügbar. Zusätzlich ist die Versorgungsspannung der Profibank auf verschiedenen Stiften herausgeführt, verbunden mit zwei sogenannten PIAs (Peripheral Interface Adapter oder Schnittstellenbaustein) des Typs 68B21, die aus der Familie des 6800, eines Vorläufers des 68000, stammen. Der preisgünstige 68B21 ist außerdem in einer stromsparenden CMOS-Version, genannt 63B21, erhältlich und läßt sich ebenfalls in der Profibank einsetzen.

Hardwarevoraussetzungen des Profiports

Jede der Leitungen *xPAy* und *xPBy* ist individuell zu einem Eingang oder Ausgang programmierbar. Jedoch unterscheiden sich die Leitungen *'PA'* hardwareseitig von *'PB'*. *'PA'*, falls zum Eingang programmiert, liegt definitiv aufgrund des internen Pull-up-Widerstands an +5V. *'PB'*, falls zum Eingang programmiert, ist hochohmig und sollte deshalb nicht offengelassen werden. Bei der CMOS-Ausführung (63B21) sind die Pull-up-Widerstände des *'PA'* konsequenterweise nicht vorhanden. Um ein sicheres Lesen der Eingänge zu garantieren, muß die Spannung an solchen für ein *'LOW'* (gelesenes Bit=0) unter 0.8V und für ein *'HIGH'* über 2.4V liegen.

'PA' des 68B21, falls zum Ausgang programmiert, erreicht von sich aus CMOS-Pegel, *'PB'* jedoch nur mit Hilfe eines Pull-up-Widerstands (Richtwert: 10K nach +5V). Der 63B21 braucht logischerweise keine Pull-up-Widerstände. Das Treiben von bis zu 2 TTL-Lasten ist in

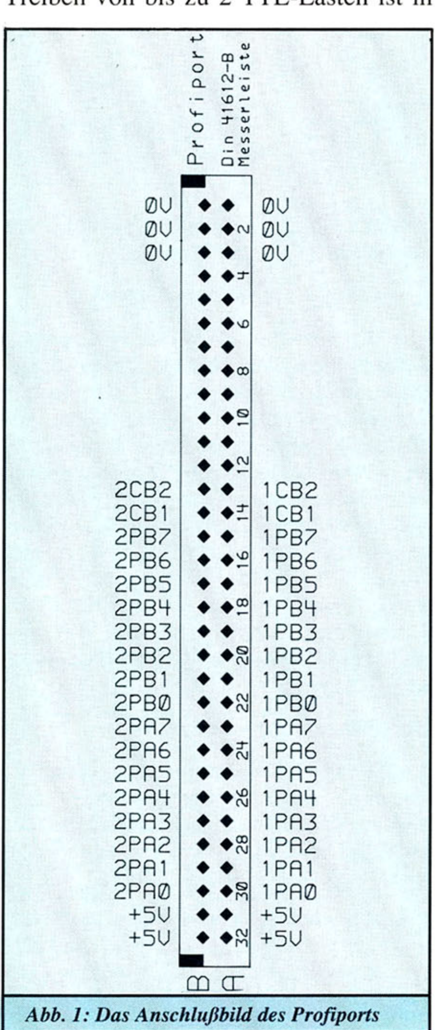


Abb. 1: Das Anschlußbild des Profiports

allen Fällen problemlos. Aufgrund ihrer Konstruktion sind die Leitungen *'PA'* zum direkten Treiben der Basen von NPN-Transistoren ohne Vorwiderstand geeignet. Die Ausgänge sind begrenzt kurzschlußfest - zu langer Kurzschlußstreß führt aufgrund exzessiver Verlustleistung eines Bausteins zu dessen sicherem thermischen Ableben.

Äußerste Vorsicht ist geboten, daß die Eingangsspannungen an den Portleitungen die Betriebsspannungsgrenzen keinesfalls überschreiten. Falls dies trotzdem geschieht, können sehr große Eingangsströme fließen, die Fehlfunktion und schließlich Zerstörung des Bausteins zur Folge haben können.

Detailliertere Informationen kann Ihnen das Datenblatt eines Herstellers liefern.

Die Programmierung des 68B21...

...gestaltet sich recht einfach, da nur 6 Register alle Funktionen abdecken. Eigentlich hat jeder Baustein nur 4 direkt erreichbare Register - denn zwei Register haben je nach Programmierung eine Doppelfunktion. Zwei Hälften, A und B, ergeben einen 68B21. Die Register im einzelnen (beachten Sie auch Abbildung 2):

1. PRA

Peripheral Data Register
oder
Datenregister

Ein Doppelregister, dessen Funktion durch das Bit 2 im Register CRA bestimmt wird.

Ist *CRA-Bit 2 = 0*, so erreicht man unter dieser Adresse das Datenrichtungsregister A, welches über Eingänge und Ausgänge der korrespondierenden Portleitungen bestimmt. Bit 7 bestimmt die Betriebsart von PA7 etc. Ist ein Bit dieses Registers Eins, so ist die entsprechende

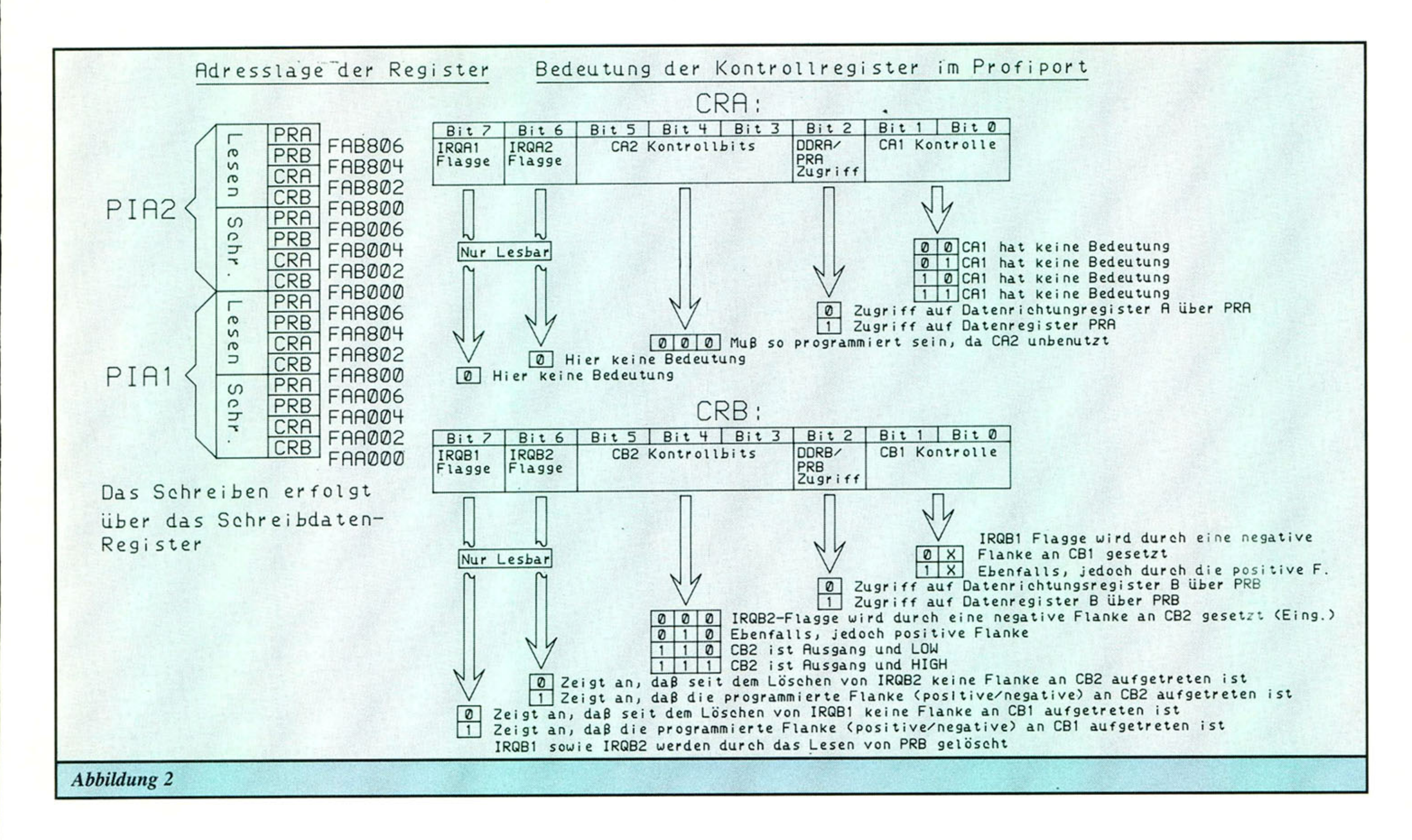


Abbildung 2

Portleitung ein Ausgang.

Falls *CRA-Bit 2 = 1*, so ist es das Datenregister. Wird es gelesen, erhält man in jedem Bit den Zustand der Eingänge und/oder der Ausgänge (je nach Programmierung des Datenrichtungsregisters). Das Lesen ergibt bei *'PA'* immer den Zustand der Portleitungen, der selbstverständlich durch äußere Beschaltung, auch bei programmierten Ausgängen, manipuliert werden kann. Ganz anders bei *'PB'*, der den Zustand eines Ausganges dem Ausgangsdatenregister entnimmt, damit auch nicht äußerlich manipulierbar ist.

Dieser Fakt erlangt spätestens dann Bedeutung, wenn Port A direkt Transistoren treibt und der Zustand eines Ausganges invertiert werden soll.

Wird das Datenregister beschrieben, gelangen die Zustände der einzelnen Bits, falls auf Ausgang geschaltet, direkt an die entsprechenden Leitungen.

2. CRA

Control Register Side A
oder
Kontrollregister A

Wie schon erwähnt, bestimmt dieses Register über den Zugriff auf das Daten- oder Datenrichtungsregister. Außerdem enthält es verschiedene Kontrollbits bezüglich der Leitungen CA1 und CA2, die jedoch im Profiport unbenutzt bleiben.

3. PRB

Wie unter 1. beschrieben, jedoch für die 2. Hälfte (Seite B) des 68B21.

4. CRB

Wie unter 2. beschrieben, jedoch für die 2. Hälfte des 68B21. Außerdem sind die Leitungen CB1 und CB2, deren Funktion dieses Register bestimmt, am Profiport verfügbar.

Abbildung 2 zeigt eine genaue Aufgliederung der Kontrollregister und die Bedeutung einzelner Steuerbits.

In Listing 1 sehen Sie das Grundgerüst mit Unterprogrammen zum Ansprechen der Profiport-Register in GFA-BASIC.

Wenn Sie eigene Hardware entwickeln ...

...zum Anschluß an die Profibank, sollten Sie folgende Richtlinien im Auge behalten, um Probleme von vornherein auszuschalten:

1. Leistungsverbraucher erhalten ein eigenes Netzteil - Steuerleitungen z.B. mit Opto-Koppler galvanisch trennen.

2. Steuern von netzspannungsführenden Stromkreisen nur mit äußerster Sorgfalt unter Einhaltung gewisser Sicherheitsrichtlinien.

3. Analoge Schaltungsteile durch ein separates Netzteil speisen, um Störeinkoppelungen zu minimieren.

4. Die Spannungsversorgung am Profiport nur zur Hilfsversorgung benutzen.

Softwareunterstützung zum Profiport

Listing 1 zeigt ein GFA-BASIC-Programm, in dem 2 Unterprogramme den Zugriff auf die Portbausteine ermöglichen: das eine Unterprogramm zum Lesen, das andere zum Schreiben eines Registers.

1. Aufruf:

@Pias (Adresse%,Datenwort%)

Dient zum Beschreiben eines Registers an der spezifizierten Adresse mit dem 8 Bit-Datenwort.

2. Aufruf:

@Pial (Adresse%,*Datenwort%)

Rückgabe des Datenwortes in *A%*

Liest das Register an der spezifizierten Adresse in die globale Variable *A%*.

Im vierten und vorerst letzten Teil unserer Profiline-Serie geben wir Ihnen eine ganze Reihe von Beispielschaltungen mit dem Profiport wie z.B. Merkkanal-Datenwandlung, Schalten und Steuern verschiedener Lasten sowie die Ansteuerung von Schrittmotoren.

RH

```
 1:  ' PIA.BAS - Beispielroutinen zur Ansteuerung des Profiports
 2:  ' R.Hofmann '88
 3:  '
 4:  Sdr%=&HFB0000! Basisadresse des Schreibdatenregisters
 5:  P1%=&HFAA000 ! Basisadresse der 1. PIA
 6:  P2%=&HFAB000 ! Basisadresse der 2. PIA
 7:  Crb%=0       ! Offset in das Kontrollregister B
 8:  Cra%=2       ! Offset in das Kontrollregister A
 9:  Prb%=4       ! Offset in das Daten- oder Datenrichtungsregister B
10:  Pra%=6       ! Offset in das Daten- oder Datenrichtungsregister A
11:  Lo%=&H800    ! Offset in den 'Lesen' - Bereich der PIA
12:  '
13:  ' ____________________________________
14:  '
15:  '
16:  ' Pias(Adresse%,Datenbyte%)
17:  ' Beschreibt das bestimmte Register mit dem Datenbyte
18:  '
19:  Procedure Pias(Adr%,Dat%)
20:    Void Peek(Sdr%+Dat%*256) ! Datenbyte in die oberen 8 Bit's des SDR
21:    Void Dpeek(Adr%)         ! In gewähltes Register Schreiben
22:  Return
23:  '
24:  ' Pial(Adresse%,*Ergebnis%)
25:  ' Rückgabewert : in Ergebnis%
26:  ' Liest das bestimmte Register und übergibt es in *Ergebnis%
27:  '
28:  Procedure Pial(Adr%,A%)
29:    *A%=Dpeek(Adr%+Lo%)/256
30:  Return
```

Listing 1: Beispielroutinen zur Ansteuerung des Profiports

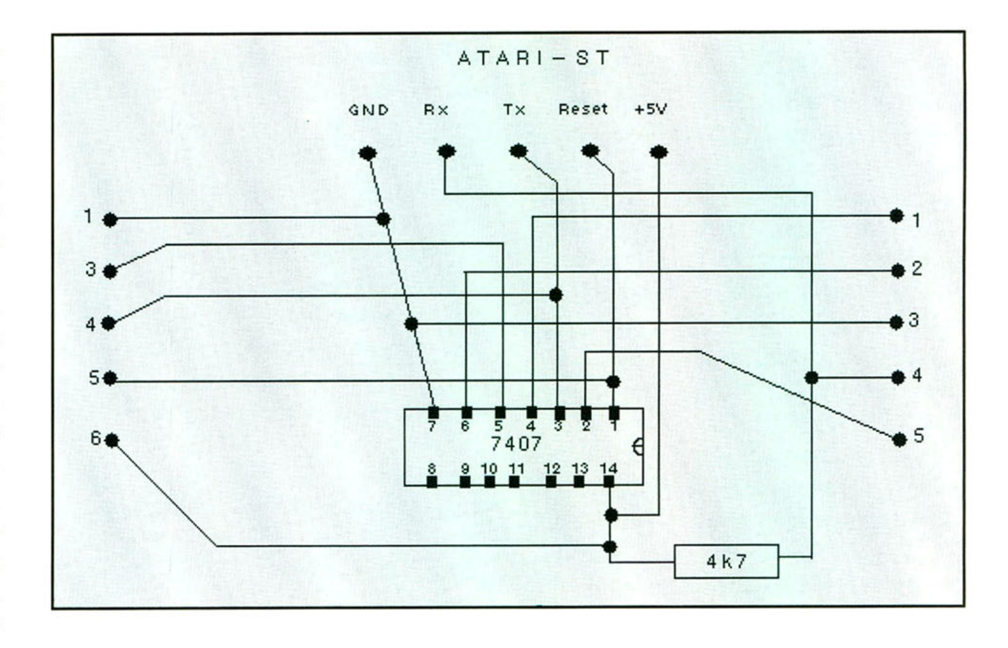
```
 1:  ' PIA.BAS - Beispielroutinen zur Ansteuerung des Profiports
 2:  ' Außerdem kann der Profiport damit getestet werden
 3:  ' R.Hofmann '88
 4:  Sdr%=&HFB0000! Basisadresse des Schreibdatenregisters
 5:  P1%=&HFAA000 ! Basisadresse der 1. PIA
 6:  P2%=&HFAB000 ! Basisadresse der 2. PIA
 7:  Crb%=0       ! Offset in das Kontrollregister B
 8:  Cra%=2       ! Offset in das Kontrollregister A
 9:  Prb%=4       ! Offset in das Daten- oder Datenrichtungsregister B
10:  Pra%=6       ! Offset in das Daten- oder Datenrichtungsregister A
11:  Lo%=&H800    ! Offset in den 'Lesen' - Bereich der PIA
12:  '
13:  ' ____________________________________
14:  '
15:  Do
16:    @Pias(P1%+Cra%,0)
17:    @Pias(P1%+Crb%,0)
18:    @Pias(P2%+Cra%,0)
19:    @Pias(P2%+Crb%,0)
20:    @Pias(P1%+Pra%,255)
21:    @Pias(P1%+Prb%,255)
22:    @Pias(P2%+Pra%,255)
23:    @Pias(P2%+Prb%,255)
24:    @Pias(P1%+Cra%,4)
25:    @Pias(P1%+Crb%,4)
26:    @Pias(P2%+Cra%,4)
27:    @Pias(P2%+Crb%,4)
28:    @Pias(P1%+Pra%,255)
29:    @Pias(P2%+Pra%,255)
30:    @Pias(P1%+Prb%,255)
31:    @Pias(P2%+Prb%,255)
32:  Loop
33:  '
34:  ' Pias(Adresse%,Datenbyte%)
35:  ' Beschreibt das bestimmte Register mit dem Datenbyte
36:  '
37:  Procedure Pias(Adr%,Dat%)
38:    Void Peek(Sdr%+Dat%*256) ! Datenbyte in die oberen 8 Bit's des SDR
39:    Void Dpeek(Adr%)         ! In gewähltes Register Schreiben
40:  Return
41:  '
42:  ' Pial(Adresse%)
43:  ' Rückgabewert : A%
44:  ' Liest das bestimmte Register und übergibt es in A%
45:  '
46:  Procedure Pial(Adr%)
47:    A%=Dpeek(Adr%+Lo%)/256
48:  Return
```

Listing 2:Mit diesem Programm ist es zusätzlich möglich, den Profiport auszutesten.

Mit viel Rubbeln ist jetzt Schluß!

Eine AT-Tastatur am ST

Die Vielschreiber unter den Lesern vermissen am ATARI-ST wohl am meisten eine 'gescheite' Tastatur. Seit der Einführung der MEGA-Serie hat sich zwar auch hier einiges gebessert, doch erstens gibt es noch Abertausende von Alt-ST-Besitzern, und zweitens ist mir persönlich der Tastenabstand auch bei den neuen ATARI-Tastaturen immer noch zu eng. Für dieses Problem haben sich verschiedene Firmen mehr oder weniger aufwendige Lösungen einfallen lassen. Da bauen welche eine komplett neue Tastatur und löten die Tasten einzeln ein. Andere sagen sich, es gibt doch eine Menge guter XT/AT-Tastaturen, kann man die vielleicht irgendwie an den ST anschließen? Man kann, und das auf mehrere Arten. Die Softwarelösungen haben den Vorteil, mit minimaler Hardware auszukommen, haben aber den Nachteil, daß sie mit einem anderen Betriebssystem als TOS im allgemeinen nicht zurechtkommen. Die Hardwarelösungen bieten zwar fast 100% Kompatibilität, kosten jedoch auch deutlich mehr. Im folgenden wird eine Hardwarelösung der Firma 3K beschrieben.



Das Produkt der Firma 3K nennt sich AT-A-ST und ist in zwei Formen lieferbar. Es besteht die Möglichkeit, den Tastaturadapter als steckfertiges Fertigprodukt zu kaufen oder aber - für Bastler besonders interessant - eine OEM-Einbauversion zu erwerben. Da mein 260ST schon vor längerer Zeit in ein PC-Gehäuse umgezogen ist, habe ich mich natürlich für die Einbauversion entschieden. Bei der Fertigversion sind keinerlei Lötarbeiten erforderlich, alles kann gesteckt werden. Der Tastaturprozessor muß herausgenommen und ein Beinchen weggebogen werden. Dann den Chip zusammen mit einem Zwischensockel wieder in die Fassung stecken. Das weggebogene Beinchen wird mittels einer Steckverbindung mit dem Zwischensockel verbunden. Dieser wiederum muß mit einem weiteren Kabel an den eigentlichen Adapter außerhalb des Rechners angeschlossen werden. An diesen werden dann auch die AT-Tastatur und das Netzteil angeschlossen, da das ATARI-Netzteil die zusätzlichen 800 mA auf gar keinen Fall aufbringen kann.

Diese Montagearbeiten müßte eigentlich jeder zustande bringen, der einigermaßen mit einem Schraubendreher umgehen kann. Die Einbauversion erfordert dagegen einen höheren Arbeitsaufwand. Hier ist auch kein Netzteil im Lieferumfang enthalten, man muß selber für eine passende Stromversorgung sorgen. Speziell für die Einbauversion ist noch eine Tastatur-Ersatzplatine erhältlich, da für die Abfrage der Maus- und der Joystickbuchse weiterhin der ATARI-Tastaturprozessor benötigt wird. Auf dieser Platine findet der Tastaturprozessor des ST sein Plätzchen, ebenso wie Maus- und Joystickbuchsen. Wer sich also von seiner treuen alten nicht trennen will, der braucht diese Platine nicht, und hat den

Vorteil, beide Tastaturen parallel benutzen zu können. Ich habe aber keinen Platz für eine zweite Tastatur, deshalb habe ich beschlossen, wenn schon AT-Tastatur, dann nur noch diese. Man muß sich zwar an ein paar Eigenheiten gewöhnen, aber einem alten ST-Benutzer dürfte das nicht so schwer fallen.

Geliefert hat 3K innerhalb einer Woche mit UPS. Das ist schon mal erfreulich schnell gegangen, ich mußte bei anderen Firmen schon wesentlich länger warten. Im Päckchen habe ich zwei Tüten vorgefunden. In der einen war der Tastaturadapter mit ein paar Tastaturkleberli und einem losen 4.7 k Ω Widerstand (wird als Pull Up für die Rx-Leitung gebraucht), in der anderen die Tastatur Ersatzplatine. Nach dem Studium der umfangreichen Anleitung (im ganzen 4 - in Worten vier - Blättern) war mir klar: Holla, das ist nicht ganz trivial.

Das fängt schon mit der mechanischen Seite an: Irgendwo im PC-Gehäuse muß ein sicheres Pätzchen gefunden werden, für die Ersatzplatine sind Ausschnitte für Maus- und Joystickanschluß herauszusägen. Auf der elektronischen Ebene sind verschiedene Stecker zu besorgen (normale 5- und 6polige DIN-Stecker, Tastaturstecker,...), und eine Treiberschaltung ist aufzubauen (alternativ, wenn keine 'lange Leitung' zwischen Tastaturprozessor und Adapter nötig ist, reicht es, wenn man auf der Adapter-Platine zwei Pull-up-Widerstände durchkneift). Natürlich sind die nötigen mechanischen Arbeiten individuell verschieden, je nachdem, wie man seinen ST eingebaut hat. Bei der Treiberschaltung frage ich mich aber, warum sie nicht schon in die Tastatur-Ersatzplatine integriert worden ist, schließlich kostet ein 7407 nicht die Welt. Wie man an dem Schaltbild sieht, ist die Schaltung an sich sehr einfach, und man kann sie ohne weiteres auf einer Lochraster-Platine aufbauen.

Gedanken muß man sich nur über die Verbindungen zum Rechner, zur Tastatur-Ersatzplatine und zum eigentlichen Adapter machen. Verdrahtet man diese Verbindungsleitungen fest, dann hat man später Probleme mit dem Aus- und Umbauen. Doch finden Sie erst mal eine 5polige Steckverbindung für die Platinenmontage. Ich habe hier etwas improvisiert, denn schließlich kriegt man mit der Metallsäge jeden Stecker in Form... . Zum Aufbau ist zu sagen, daß man unbedingt isolierte Kupferlitze (lackierter, dünner Kupferdraht) verwenden sollte, um einen Drahtverhau zu vermeiden. Leider hatte ich grad' keine zur Hand, deshalb bietet mein Aufbau keinen sehr ästhetischen Anblick, wenn man ihn von der Lötseite her betrachtet. Nun ja, immerhin funktioniert er.

So, nachdem nun auch der ST restlos verkabelt ist, muß man nur noch den Tastatur-Prozessor aus der alten ST-Tastatur vorsichtig mit einem Schraubendreher raushebeln und in die Ersatzplatine einstecken. Bei allen Stöpselvorgängen empfehle ich die großzügige Verwendung von einem Videospray, um Fett und sonstigen Dreck zu entfernen. Nachdem man noch einmal alle Verbindungen und den richtigen Sitz der ICs überprüft hat, steht einem ersten Test nichts mehr im Weg. Zuerst soll man das AT-A-ST-Netzteil einschalten, dann den ST. Die Maus muß ganz normal funktionieren, denn schließlich werkelt hier der original ATARI-Tastatur-Prozessor. Spannend wird es dagegen beim Ausprobieren der AT-Tastatur. Nach Angaben von 3K gibt es Probleme mit einigen Taiwan-Tastaturen. Das liegt vermutlich daran, daß die serielle Datenübertragung bei der ATARI-Tastatur asynchron erfolgt, bei der AT-Tastatur jedoch synchron. Synchrone Übertragung bedeutet, daß die AT-Tastaturen neben den Daten noch ein Taktsignal produzieren. Die Geschwindigkeit der Datenübertragung wird durch diesen Takt bestimmt und ist deshalb in einem weiten Bereich variabel. Einige Taiwan-Hersteller verwenden nun sehr unübliche Taktfrequenzen - anscheinend waren diese Quarze besonders billig zu haben. Dadurch könnte es zu den Problemen kommen.

Geschickterweise habe ich mir zuerst eine PC-Tastatur gekauft, und mich erst dann um den AT-A-ST gekümmert. Dreimal dürfen sie nun raten, woher diese stammt: Richtig, aus Taiwan! Noch dazu war sie umschaltbar zwischen XT und AT-Betrieb, was noch einmal besonders kritisch sein soll. Dagegen funktionieren die reinen AT-Tastaturen und die Auto-Switch-Tastaturen einwandfrei - nach Aussage von 3K. Im Bekanntenkreis habe ich nur noch eine Tastatur auftreiben können, die - wie meine - manuell umgeschaltet werden konnte. Nun, um es kurz zu machen: Beide funktionieren - mit einer kleinen Macke. Bei 1st_Word kommt es beim Scrollen des öfteren vor, daß der Cursor Zeilen überspringt. Das ist weiter nicht tragisch, es kommt zu keinem Datenverlust. Trotzdem sollte man bei der Auswahl der Tastatur aufpassen und gegebenenfalls bei 3K nachfragen.

Details

Noch ein paar Worte zum Adapter selbst. Er besteht aus einem Z80-Einplatinen-Computer und einer Platine, die die Anpassung synchron/asynchron vornimmt. Beide Platinen werden huckepack-förmig zusammengesteckt. Der Z80-Rechner ist keine Eigenentwicklung der Firma 3K, sondern wird - zumindest die Platine - vom Elektronikladen in Detmold bezogen. Dagegen wird die Tastatur-Ersatzplatine vollständig von 3K hergestellt.

Verarbeitung Ausstattung

Grobe Mängel habe ich nicht festgestellt, alles hat auf Anhieb funktioniert. Doch gibt es auch einige Kritikpunkte. Die Platinen, die 3K selbst hergestellt hat, machen keinen professionellen Eindruck, wenn man sie zum Beispiel mit der Platine des Z80-Rechners vergleicht. Nun, das muß nicht unbedingt schlecht sein, doch hätte man vielleicht ein bißchen Lötlack opfern können, damit die Platinen versiegelt sind. Zwar sollte man als Bastler so etwas im Haus haben, aber nett wäre es trotzdem. Unangenehmer finde ich, daß man die bereits angesprochene Treiberschaltung selber aufbauen muß. Bei der Fertigversion ist sie in den Zwischensokkel für den Tastaturprozessor integriert. Warum nicht auch bei der Tastatur-Ersatzplatine? A propos Ersatzplatine, hier hat mir sehr mißfallen, daß die Joystickbuchsen zwar Befestigungslöcher aufweisen, diese aber nicht benutzt werden. Die Buchsen sind einfach nur eingelötet, was bei häufigem An- und Abstöpseln bestimmt zu Problemen führen wird (einen Bruchtest habe ich - aus naheliegenden Gründen - nicht durchgeführt). Ich für meinen Teil werde die Buchsen jedenfalls mit Zweikomponentenkleber zusätzlich sichern.

Mit der Anleitung bin ich zwar zurechtgekommen, doch das Gelbe vom Ei ist sie sicher nicht. Wobei ich mich jetzt auf die Einbauversion beziehe, die Anleitung für die Fertigversion ist mit 1 Seite zwar auch recht knapp, aber sie müßte ausreichen, den Adapter in Betrieb zu nehmen. Hervorzuheben ist die telefonische Hotline, die man von Montag bis Freitag von 17-19 Uhr in Anspruch nehmen kann. Mit Hilfe dieser Hotline konnte ich Fragen, die durch die Anleitung offen geblieben waren, schnell abklären.

Billig ist der Spaß auch nicht:

AT-A-ST Fertigversion	298.- DM
OEM-Einbauversion	198.- DM
Tastatur-Ersatzplatine	79.- DM
AT-Tastatur	120.- DM
bis über	500.- DM

Insgesamt muß man also mit deutlich über 400.- DM rechnen, auch wenn man eine billige Tastatur kauft. Meine gehört mit ca. 160.- DM in die untere Preiskategorie, doch bin ich sehr zufrieden mit ihr. Der Anschlag ist wesentlich besser als auf meiner alten ST-Tastatur, und, bedingt durch den weiteren Tastenabstand, vertripepe ich mich weit weniger. Kurzum, ein wahrer Segen für Vielschreiber.

Das Wort zum Schluß

ST-Benutzer, die nur unter TOS arbeiten, sollten sich die Software-Lösungen anschauen, die ein ganzes Stück billiger sind. Wer dagegen mehrere Betriebssysteme (PC ditto, Aladin, Mirage, RTOS, OS/9, ...) verwendet, der wird sich entweder selber etwas stricken müssen, oder wenn er Glück hat, gibt es bereits eine Anpassung. Oftmals wird hier aber eine reine Hardwarelösung nötig sein, um zu befriedigenden Ergebnissen zu kommen. Der AT-A-ST von 3K bietet nun 100% Kompatibilität - von einigen Eigenheiten der AT-Tastatur einmal abgesehen. Mit der Fertigversion dürfte auch der Laie zurechtkommen, wogegen die Einbauversion eher für Bastler mit ein wenig Erfahrung geeignet ist.

Verbesserungsfähig sind die Anleitung und die mechanische Befestigung der Joystickbuchsen. Außerdem könnte man das Treiber-IC noch in die Tastatur-Ersatzplatine integrieren. Vielleicht ließe sich auch eine steckbare Verbindung der Einzelbaugruppen untereinander verwirklichen, denn für den Privatmann ist die Beschaffung von geeigneten Steckern zuweilen schwierig.

Bei mir funktioniert die Einbauversion zusammen mit der Tastatur-Ersatzplatine und einer AT-Taiwan-Tastur einwandfrei. Der hohe Preis wird jedoch viele potentielle Käufer abschrecken.

C.D. Ziegler

Bezugsnachweis:

3K EDV-Entwicklungen
Hülser-Straße 76
4154 Tönisvorst



A. Büdenbender · ☎ 02 21 / 430 14 42 · FAX 46 65 15
NEUE ADRESSE jetzt in der Mommsenstr. 72 / Ecke Gleulerstr.
IHR FACHHÄNDLER IN KÖLN FÜR ATARI / PC / AT
Wir bieten Ihnen noch Beratung und Service für Ihren Computer

MONITORE

EIZO Multimonitor 800★600 Drehfuß 0.28 Dot dt. Version 7 Monate Garantie 3 Aufl.	1548,–
NEC Multisync GS alle drei Auflösungen an St Serie	538,–
Profex Farbmonitor Scart	598,–
SM 124 Atari Monitor	430,–
Matrix Monitor 19 Zoll	5200,–

Abgesetzte Tastatur am ST

Tastatur Gehäuse mit Schalter, Spiralkabel, Joystickbuchsen, Treiber	130,–
AT Tastatur am ST mit Interface für Rom Port und AT Tastatur 102 Tasten	298,–

Speicher Karten ST

Speicherkarte 512kb mit Rams steckbar von CSF ST260/520	380,–
Leerkarte 4 MB St 1040/St260	298,–
Juniorprommer von Merlin	189,–

Kabel div. ST

Scart Kabel fertig 3m/1,5m	49,–/39,–
Modem Kabel 2 Meter	29,–
Drucker Kabel Centronics 2 Meter	25,–
Floppy Kabel Lw. A .... 25,– .... A+B	30,–

Festplatten ST

Vortex 30 MB Platte HD30+	1298,–
Vortex 20 MB Platte HD20+	1100,–
Vortex 60 MB Platte HD60+	1998,–
Atari 30 MB Platte NEU	1300,–
Atari 60 MB Platte NEU	1850,–
Wechselplatte 20 MB für St	2498,–

Alle Festplatten mit Turbo Dos Vortex mit Backup

Drucker / Laser

NEC P6 Plus dt. Version 12 Mon. Garantie	1698,–
NEC P2200 NEU 24 Nadeln	899,–
STAR LC 24/10 24 Nadeln 154 Sek.	1098,–
STAR LC 10 Drucker 9 Nadeln dt. Ausf.	620,–
ATARI Laserdrucker SLM 204 2 Seiten Min.	3000,–
Panasonic 1540 DIN A3 24 Nadeln 240 Z.	1800,–
Panasonic 24 Nadeln P6 kompatibel 160 Z.	1050,–
Epson LQ 500 24 Nadeln komplett	950,–

AUTO Switchbox

Switchbox mit Software 40 cm Kabel BAS Audio Ausgang	59,–
Switchbox o. Software mit Schalter, mit Reset beim Umschalten	45,–
Switchbox für Multisync GS Eizo/NEC ohne Reset mit Software	69,–

Die Switchboxen mit Software erkennen bei der Umschaltung mit Mullmonitor Farbe/schwarzweiß automatisch. Sie brauchen nicht mehr umschalten.

FLOPPY Laufwerke ST

3.5 Zoll und 5.25 Zoll Laufwerk bester Qualität NEC und Teac NEU auch KOMBI 3.5 und 5.25 Zoll als B/A Lw. die Profi Laufwerke 3.5 Zoll mit Bus für weitere Lw. externes Steckernetzteil

ST Floppy 3.5 Zoll mit Bus für weiteres Lw. externem Netzteil VDE NEC FD1037 726KB abschaltbar	299,–
ST Floppy 3.5 Zoll ohne Bus sonst wie oben FD 1037	249,–
ST Floppy mit Trackanzeige 3.5 Zoll ohne Bus abschaltb.	349,–
ST Floppy 5.25 Zoll internem Netzteil 40/80 Teac kompatibel zu PC Ditto / PC Disk abschaltbar Anschlußfertig	369,–
ST Floppy 3.5 Zoll + 5.25 Zoll in einem Geh. A/B schaltbar	648,–

Hardware Zubehör

UHR St Steckbar intern bei Blittertos keine Software mehr	105,–
Rom Satz ST .. 98,– Eproms 27256 .. p. Stk.	14,–
Floppy FD 1037 .. 179,– Steckernetzteil 5V	30,–
Floppy FD 1036 .. 169,– Netzteil 12V / 5 V	40,–

Computer ST / Mega ST

Atari ST 1040 mit SM 124 Monitor 10 Disk	1598,–
Atari Mega St 2 mit SM 124 Monitor mit Blitter Tast.	2650,–
Atari Mega St 4 mit SM 124 Monitor mit Blitter Tast.	3600,–
Atari Mega St 4 mit Laser SLM 804 und Scanner Panasonic FX RS 505 mit Calamus Software	9999,–
Scanner mit 400★400 Dpi Anschlußfertiges System Scanner Panasonic Fx 505 Anschlußfertig	3198,–

Telefax Geräte

Schneider Telefax für Ihr Büro	2400,–
Canon Telefax 230 16 Graustufen 60 Rufn.	3950,–
Panasonic UF 150 Fax 16 Graustufen 60 Rufn.	3200,–

Disketten Freesoftware

Fuji Disk 1DD 3.5 Zoll ab 30 Stk pro 10 Stk.	27,–
Fuji Disk 2 DD 3.5 Zoll je 10 Stk.	35,–
Freesoftware alle Prg. aus ST Computer pro Disk .... 8,– bei 10 Disketten	60,–

Wir liefern für Ihre Firma die richtige Soft/Hardware Beratung nach Wunsch

Atari / Star NEC Schneider/Atari/Epson sind eingetragene Warenzeichen
Versand ins Ausland nur per Vorkasse

Öffnungszeiten im Ladengeschäft:
10 – 13 Uhr – 14 – 18 Uhr – Samstag 10 – 14 Uhr.

Alle Preise in DM

Adimens 2.1 Datenbank	185,–	Signum 2	388,–	Stad Grafik	159,–	PC Ditto	179,–
ST Pascal 2.00 plus CCD	249,–	GFA Basic 3.02	179,–	BTX ST	298,–	1St Word+	180,–
Star Writer ST Textprogr.	189,–	Tempus 2.0	110,–	AdiTalk St	190,–	1St Adress	135,–
Spooler ...... 39,– Campus Draft	158,–	Campus Art	158,–	Fibu Man E	369,–	BS Handel	450,–

Panasonic OA FX-RS505 (ST)

Flachbettscanner für ATARI Mega ST und MS-DOS Rechner



- -wählbar bis 400 x 400 dpi
- -anpassbarer Kontrast
- -Line- und Halbtonbetrieb
- -Schnittstellen inbegriffen

MS-DOS Rechner -Version:
DM 3.498,--

ST -Version incl. Grafiksoftware:
DM 3.798,--

Distribution:

COMPUTERSTUDIO
Werner Brock GmbH
Untere Gerberstr. 15
7410 Reutlingen
Tel: 07121/34287, Telefax: 07121/339779



DAS GESCHENK

Nicht dokumentierte Befehle des OMIKRON.BASIC 3.00

Bei einem tieferen Blick mit einem Monitor in das neue OMIKRON.BASIC 3.00 kam es an den TAG: OMIKRON.BASIC kann mehr als im Handbuch angegeben. Was lag also näher, als diesen undokumentierten Befehlen auf den Grund zu gehen?

Die Rücksprache mit der Herstellerfirma ergab, daß diese Befehle zwar eingebaut aber bislang zurückgehalten wurden, da sie noch nicht 100%ig ausgetestet waren, und OMIKRON.Software dem Kunden nur fehlerfreie Befehle präsentieren will.

Um die Spannung nicht allzu groß zu machen - Überraschungen wird es in diesen Monat eh noch genug geben -, beginne ich am besten gleich mit dem ersten Befehl.

```
NDC [<X>,<Y>,<W>,<H>]
NDC [<X>,<Y> TO <W>,<H>]
```

NDC bewirkt die interne Umrechnung von Grafikbefehlen (z.B. BOX, CIRCLE,..) auf *normalisierte* Koordinaten (0..32767).

Mit Hilfe dieses Befehls können Programme, die Graphik ausgeben, im Programm mit festen Werten arbeiten und trotzdem auf allen Monitortypen lesbare Ausgaben erzielen. Dazu ist es nur am Programmanfang nötig, die absolute Auflösung zu erfragen, und mit NDC die Umrechnungskoordinaten zu setzen.

Beispiel:

```
'    NDC
' s/w und Farbe
Maxint%=32767
XBIOS (Res,4)'          Auflösung holen
Xmax=639+320*(Res=0)'        maximale Koordinaten
Ymax=399+200*(Res<2)'        errechnen
NDC 0,0,Xmax,Ymax'NDC einschalten
CLIP 0,0,maxint%,maxint%
'
DRAW 0,0 TO Maxint%,Maxint%
DRAW 0,Maxint% TO Maxint%,0
```

Es erscheinen die zwei Diagonalen auf dem Bildschirm, und zwar in jeder Auflösung. NDC ohne Parameter macht die Einstellung wieder rückgängig und kehrt zum normalen Koordinatensystem zurück.

Es folgt ein etwas komplexeres Beispiel des NDC-Befehls, wo zunächst die Auflösung erfragt und daraus die maximalen X- und Y- Koordinaten errechnet werden.

Anschließend berechnet das Programm eine Sinus-Welle und legt die Werte in einem Feld ab. Dieses wird sodann mehrfach geplottet, aber immer mit unterschiedlichem NDC-Befehl. Es zeigt, daß es mit NDC eine Leichtigkeit ist, Programme so zu schreiben, daß sie in jeder Auflösung lauffähig sind.

Falls sie schon einmal einen Blick in einschlägige GEM-Literatur geworfen haben, werden Sie von NDC (*Normalized Device Koordinates*) schon gehört haben, doch im GEM waren sie bislang nicht funktionsfähig - OMIKRON.-BASIC macht's möglich. Im übrigen ist die gesamte GEM-LIB 'NDC'-fähig.

```
'    NDC
' s/w und Farbe
Maxint%=32767
XBIOS (Res,4)' Auflösung holen
Xmax=639+320*(Res=0)' maximale Koordinaten
Ymax=399+200*(Res<2)' errechnen
Xy!=Ymax/Xmax'      Faktor zwischen x und y
DIM X(Xmax),Y(Xmax)

FOR I=0 TO Xmax' Funktionswerte errechnen
  X(I)=I*Maxint%/(Xmax)
  Y(I)=Maxint%/2+ SIN(I/Ymax*10)*Maxint%/2
NEXT I


FOR X=0 TO Xmax/2-1 STEP 10
  CLS
  Y=X*Xy!:B=Xmax-2*X:H=B*Xy!
  NDC X,Y,B,H' <—— neues Koordinatensystem
  CLIP 0,0,maxint%,maxint%
  BOX 0,0 TO Maxint%,Maxint%' Rahmen ziehen
  FOR I=0 TO Xmax
    DRAW X(I),Y(I)' Funktion zeichnen
  NEXT I
NEXT X
```

Es gibt aber noch eine andere Möglichkeit, Programme für alle Auflösungen zu schreiben. Im vorigen Beispiel wurden nur die normalen ST-Auflösungen berücksichtigt - da gibt es nun aber beispielsweise den *Matrix-Screen*, der wesentlich mehr Punkte auflösen kann. Damit Ihre neuen Programme auch in Zukunft auf allen Monitoren laufen, holt man sich die maximalen Koordinaten per *Wind_Get*. Praktisch hierbei ist, daß die Menüleiste gleich ausgeschlossen wird. Probieren Sie das folgende Beispiel einfach einmal aus (siehe unter "NDC" auch für den Großbildschirm).

POLYGON, PPOLYGON

Diese Befehle sind bekannt aus

```
' NDC auch für Großbildschirm
'
LIBRARY Gem ,"f:\omikron\gem.lib"' GEMLIB laden
Maxint%=32767
Appl_Init'                  Rein ins GEM
Wind_Get(0,4,X,Y,B,H)' Maximale Ausmaße holen
NDC X,Y,B,H'                NDC-Koordinaten
CLIP 0,0,Maxint%,Maxint%'   Clip mal wieder
PRINT : PRINT X,Y,B,H'      Werte mal ausgeben
DRAW 0,0 TO Maxint%,Maxint%' Diagonalen
DRAW 0,Maxint% TO Maxint%,0' zeichnen
WAIT 10'                    Kurze Pause
Appl_Exit'                  Raus aus GEM
LIBRARY CODE Gem
```

dem VDI, doch ist es allemal bequemer, sie direkt im BASIC aufrufen zu können.

POLYGON und PPOLYGON dienen zum Zeichnen von Linienketten, wobei PPOLYGON diese Fläche nach einem bestimmten Schema füllt.

Zwar kann man mit mehreren DRAW-Anweisungen diesen Befehl emulieren, doch spätestens beim Versuch, ausgefüllte Flächen in einer akzeptablen Zeit auf den Bildschirm zu zaubern, dürften die meisten Programmierer Ihre Schwierigkeiten haben.

Die Eckdaten des Polygons werden zu diesem Zweck in ein zweidimensionales Integer-Feld abgelegt, dessen erster Index zweidimensional ist, der zweite der Anzahl der Eckpunkte entspricht.

Ein beispielhafter Aufruf *[P]-POLYGON A%(0,8)* zeichnet ein nicht unbedingt geschlossenes Vieleck - den abgelegten Koordinaten von *A%(0,0)* bis *A%(1,8)* entsprechend, wobei in *A%(0,_)* die X-Koordinaten und in *A%(1,_)* die Y-Koordinaten verlangt werden.

OUTLINE {ON/OFF}

entscheidet, ob ein Grafikobjekt (z.B. PCIRCLE, PBOX,..) umrahmt oder rahmenlos gezeichnet wird.

```
OUTLINE OFF - ohne Rahmen
OUTLINE ON - mit Rahmen
```

BITBLT

Zwar existiert dieser äußerst leistungsstarke Befehl schon immer, doch hat er einige Features hinzubekommen.

Neu ist die Berücksichtigung von Clippingrechtecken und die mögliche Verknüpfung von Speicherbereichen, wenn das Ziel der Speicher ist, also nicht nur beim Kopieren vom Speicher in den Bildschirmbereich. Sollte ein Blitter installiert sein, wird dieser angesprochen, falls nicht, wird nicht die *BITBLT*-Routine des Betriebssystem angesprungen, sondern die eigene, wesentlich schnellere Routine des Interpreters.

Die Syntax lautet wie bisher:

```
BITBLT <X>,<Y>,<W>,<H>
   TO <MEMORY> ; <MODUS>
```

Beispiel:

```
' POLYGON PPOLYGON
' Beispiel s/w und Farbe
XBIOS (Res,4)' Auflösung erfragen
Xmax=639+320*(Res=0)' maximale Koordinaten
Ymax=399+200*(Res<2)' errechnen
CLIP 0,0 TO xmax,ymax

N=10' Anzahl der Polygon-Ecken
DIM P%(1,N)'Integer-Feld anlegen
CLS 'nicht nur sauber, sondern pixeltief rein
FOR I=0 TO N
  P%(0,I)= RND(Xmax)' X-Wert erzeugen
  P%(1,I)= RND(Ymax)' Y-Wert erzeugen
NEXT I
PPOLYGON P%(0,N)' Polygon zeichnen
```

JOYSTICK

Damit man auch, ohne den Tastaturprozessor in Eigenregie quälen zu müssen, die Joystickstellung erfragen kann, wurde eine neue Funktion implementiert.

```
JOYSTICK(<Nummer>)
```

Nummer steht für die Nummer des Joystickports, wobei 0 für den Maus-, 1 für den Joystickport steht. Bit 7 des Rückgabewertes gibt den Status des Feuerknopfes, Bit 0-3 die Richtung an. Um die Abfrage zu aktivieren, muß dem Tastaturprozessor mitgeteilt werden, daß er ab sofort Joystickdaten senden soll. Das Beispielprogramm verdeutlicht die Abfrage:

```
' JOYSTICK
BIOS (,3,4,20)' Abfrage ein
REPEAT
  PRINT JOYSTICK(1),
  PRINT JOYSTICK(0)' Bitte Maus raus !!!
UNTIL INKEY$ <>""' bis Tastendruck
BIOS (,3,4,8)' Abfrage aus
```

Nach der Abfrage muß der Tastaturprozessor wieder auf Normalmodus geschaltet werden, da sonst die Maus inaktiv bleibt (siehe letzte Programmzeile).

Sie erhalten folgende Werte:

5	1	9
4	128	8
6	2	10

INLINE

Zum Einbinden kurzer Maschinenroutinen steht nun der INLINE-Befehl zur Verfügung. Der Maschinencode wird in einem einfachen String angegeben. Die Länge ist dabei allerdings auf 124 Byte begrenzt.

INLINE hat den großen Vorteil, daß der Befehl selbst alle Register rettet (auch der RTS am Ende kann entfallen). Das INLINE-Programm kann mit ihnen also sorglos umgehen - der Stack-Pointer allerdings sollte wieder korrigiert werden.

Das folgende Beispiel beinhaltet einen INLINE-Aufruf, der den gesamten Bildschirminhalt achtmal um ein Pixel nach oben schiebt. Man erhält somit einen wunderbaren Softscroll, (Das Beispiel für den INLINE-Aufruf ist auf der nächsten Seite zu sehen).

Dateibehandlung

In der Dateibehandlung hat sich auch einiges getan. GEMDOS-Programmierer werden mit den neuen Befehlen wenig Schwierigkeiten haben, denn sie sind in der Funktion mit den entsprechenden GEMDOS-Aufrufen identisch, nur sind sie in BASIC-Syntax eben leichter anzuwenden. Im folgenden soll die Syntax gezeigt werden, die genauen Funktionsweisen müssen Sie bei den GEMDOS-Funktionen nachschlagen (z.B. im OMIKRON.Handbuch).

```
OPEN "U",<Filenr>,<Filename>
```

eröffnet eine Datei für folgende Befehle

```
SEEK<Filenr>,<Pos>[,Mode>]
```

setzt den Dateizeiger auf die angegebene Position

```
GET <Filenr>,<Addr>,<Len>
```

holt *Len-Bytes* aus Datei und speichert im Speicher ab Adresse *Addr*

```
GET <Filenr>,<String>,<Len>
```

liest in String z.B. *GET 1,-A$,100*

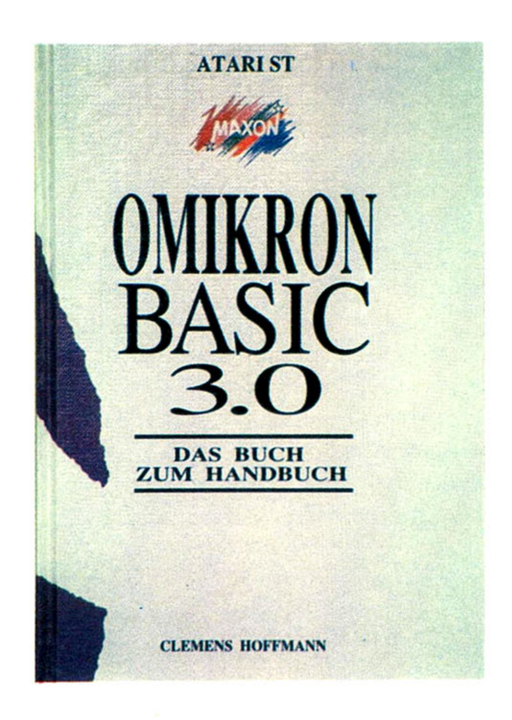
```
PUT <Filenr>,<Addr>,<Len> bzw.
PUT <Filenr>,<String>
```

schreibt in die Datei z.B. *PUT 1,MID$(A$,3,97)*

Die Bedeutung der Variablen Pos ist wie folgt:

Pos=0: Position wird vom Dateianfang gezählt (entspricht Aufruf ohne Mode)

OMIKRON.BASIC 3.0
Das Buch zum Handbuch



Aus dem Inhalt des Buches

Für den absoluten Neuling

Kurze Einführung in die BASIC-Programmierung.

Über das Handbuch hinausgehende Beschreibung vieler Befehle, Besonderheiten und Kniffe. Verwendung selbstdefinierter Prozeduren und Funktionen. Viele Beispiele, Aufgaben mit Lösungen.

Für den Aufsteiger, aber auch für den geneigten Anfänger:

OMIKRON.Sprites-Tücken, Vorteile, Anwendung.
Overlay-Technik (Auslagern langer Programmteile und Laden bei Gebrauch).

Grundlagen der strukturierten Programmierung. Schreiben eigener und Verwenden fremder Libraries (Bibliotheks-Funktionen).

Aufrufe von TOS und GEM im BASIC (GEMLib); endlich die Wahrheit über die GEM-Aufrufe!

Dabei wird auch das GEM-Zusatzprogramm GDOS berücksichtigt.

Sound und Grafik-Programmierung. Grafische Effekte (z.B. die Verwendung mehrerer Grafik-Bildschirme und Zeichnen in nicht sichtbare Bildschirme. Aufbau von Metafiles, IMG-Bildern u.a.

Aufbau und Verwenden der Menü-Leisten in GEM-Accessories in OMIKRON.BASIC.

Verwendung der BASIC-internen Multitasking-Befehle.

Einige Libraries (Turtle-Grafik Erweiterungen und Korrekturen zur GEMLib, usw).

Erklärung der Befehle der Version 3.0 und ihre Anwendung. Die Feinheiten des Compilers V 2.0.

Natürlich befinden sich alle Programme und Beispiele auf Diskette!



Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: ______________________

Vorname: ______________________

Straße: ______________________

Ort: ______________________

Unterschrift: ______________________

Hiermit bestelle ich:

☐ Exemplare von "OMIKRON BASIC 3.0". Mit Diskette für DM 59,00

Versandkosten: DM 7,50
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr.

☐ Vorauskasse
☐ Nachnahme

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3

Pos=1: Distanz relativ. Wert (positiv oder negativ) geht um angegebenen Wert vor oder zurück.

Pos=2: Position wird vom Dateiende gerechnet

Das etwas komplexere Beispielprogramm lädt ein Bild im alten DEGAS-Format auf den Bildschirm. Dabei wird ebenfalls die Farbpalette berücksichtigt.

DEGAS-Format ist folgendermaßend aufgebaut:

```
    2  Byte Auflösung
 16*2  Byte Farbinformation
32000  Byte Bildinfo
```

Zum Auslesen der Farbinfo wird also der Dateizeiger auf 2 gesetzt, zum Einladen des Bildes auf 34.

Um in Programmen mit Speicherbereichen spielen zu können, sind vier neue Befehle

```
'OPEN "U"
'
XBIOS (Bildschirm,2)
OPEN "U",1,"A:\MOUNTAIN.PI3"' DEGAS-Pic
IF LOF(1)=32034 THEN
  DIM C(16)
  SEEK 1,2' Zeiger auf Farbinfo
  GET 1,Farb$,32'  Farbpalette lesen
  FOR C=0 TO 15
    C(C)= CVI( MID$(Farb$,C*2+1,2))' AND $777
    PALETTE C(0),C(1),C(2),C(3),C(4),C(5),C(6),C(7),C(8),
            C(9),C(10),C(11),C(12),C(13),C(14),C(15)
  NEXT C
ENDIF
SEEK 1,34' altes DEGAS-Format

GET 1,Bildschirm,32000' Bildinfo laden
CLOSE 1
REPEAT UNTIL INKEY$ <>""
'PALETTE $777,0'  falls ungünstige Farbkombination
```

MEMORY_MOVE, MEMORY_MOVEB

hinzugekommen. Die ersten zwei dienen der Speicherverschiebung, nicht mit BITBLT zu verwechseln, wobei der einzige Unterschied zwischen ihnen ist, daß der eine wortweise, daher schneller, und der andere bytebezogen arbeitet.

```
MEMORY_MOVE bzw.
MEMORY_MOVEB<Start>,<Länge>
TO <Ziel>
```

Bei _MOVE ist auf drei gerade Werte zu achten, da sich der Prozessor sonst beschwert.

MEMORY_BLOCK

Dieser Befehl ähnelt MEMORY, doch wird der Speicher nicht vom GEMDOS reserviert, sondern direkt im BASIC-Programm angelegt. Somit können Daten mit dem Programm abgespeichert werden.

```
MEMORY_BLOCK
<Nr.>,<Länge>,<Adresse>...
```

...reserviert soviele Bytes, wie in Länge angegeben sind, und übergibt der Variablen Adresse die Adresse, über die auf den Block zugegriffen werden kann. Nummer muß eine zweistellige Zahl (00-99) sein. Es können also bis zu 100 Blöcke definiert werden. Eine Anwendung wäre beispielsweise das direkte Einbinden von Sprite-Daten, sie müßten dann nicht mehr nachgeladen oder in DATA-Zeilen abgelegt werden.

Die Adresse eines Datenblocks liegt nicht fest, absolut adressierende Maschinenprogramme können auf diese Art nicht eingebunden werden.

Da das Erzeugen eines solchen Blocks etwas komplizierter ist, empfehlen wir folgendes Schema:

```
MEMORY_BLOCK
<Nr>,<Länge>,<Adresse>
```

Zunächst mit einen Block reservieren

```
' INLINE
' Beispiel für s/w und mittlere Auflösung

PRINT CHR$(27);"f"' Cursor aus
BLOAD "c:\utility\joshua\joshua.pic"' 32000K-Bild laden
'Falls kein Bild vorhanden, tut's auch Text
'FOR I=0 TO 23: PRINT "O wie soft ist mein Scroll": NEXT I

FOR I=0 TO 24
  INLINE "3f3c00024e4e548f343c000f323c03e5204043e800504
          cd938f848d038f841e8002051c9fff251caffe44e75"
  ' 8 Pixel nach oben scrollen
NEXT I


start:   move.w  #$2,-(sp)          ;3f3c 0002
         trap    #$e                ;4e4e
         addq.l  #$2,sp             ;548f
         move.w  #$f,d2             ;343c 000f
loop1:   move.w  #20*399/8,d1       ;323c 03e5
         movea.l d0,a0              ;2040
         lea.l   $50(a0),a1         ;43e8 0050
loop:    movem.l (a1)+,d3-d7/a3-a5  ;4cd9 38f8
         movem.l d3-d7/a3-a5,(a0)   ;48d0 38f8
         lea.l   $20(a0),a0         ;41e8 0020
         dbra    d1,loop            ;51c9 fff2
         dbra    d2,loop1           ;51ca ffe4
ende:    rts                        ;4e75
```

```
BLOAD "Dateiname",Adresse:STOP
```

Die Daten werden in den Block geladen und das Programm gestoppt.

Danach die BLOAD-Zeile entfernen oder mit REM versehen. Der Datenblock befindet sich nun im BASIC-Code.

FRE

Bei Verwendung des klassischen MEMORY-Befehls fehlte bislang die Möglichkeit, diesen wieder freizugeben. Der Befehl

```
FRE(<Adresse>)
```

erledigt dies prompt. Adresse muß die Adresse sein, die beim Memoryaufruf zurückgegeben wurde. Es sollte aber beachtet werden, daß GEMDOS nicht beliebig viele Memoryaufrufe verkraften kann.

SORT

Eine kleine, doch sehr nützliche Änderung wurde im Sort-Befehl realisiert. Häufig kam es vor, daß man in einem hoch dimensionierten Feld nur wenige benutzte. Ein folgender SORT-Befehl nahm keine Rücksicht auf die leeren Felder.

Mit

```
SORT <Feldname>(<Anzahl>)
```

ist es nun möglich, nur bis zum Index (Anzahl-1) zu sortieren. Beträgt der Parameter Anzahl Null, so wird das gesamte Feld sortiert.

L

Bei den Aufrufen von GEMDOS, BIOS, XBIOS oder CALL wurde bislang die Aufteilung in HIGH- und LOW-Byte gefordert - nun sind auch Langworte erlaubt. Dem übergebenen Wert wird dabei ein *L* vorausgestellt. Das Beispiel lädt den Bootsektor der Diskette ein, stellt ein *RTS ($4E75)* an den Anfang und schreibt ihn wieder zurück. Es handelt sich hierbei um einen einfachen Viruskiller, der beim Booten bewirkt, daß das Bootprogramm nicht ausgeführt wird. Wenden Sie es nie bei TOS-Disketten oder selbststartenden Programmen (meist Spiele) an!!! (Siehe Beispiel MINI-VIRUS-KILLER OMIKRON auf der nächsten Seite).

LIBRARY

Als geradezu revolutionär kann man den LIBRARY-Befehl bezeichnen. Er ermöglicht

das Einbinden einer speziell gepackten Bibliothek in das eigene Programm. Als Beispiel kann man die neue *GEM-Lib* der Version 3.00 betrachten. Bislang mußte sie 'eingemergt' werden, was erstens längere Zeit in Anspruch nahm (Wer hat die schon?) und zweitens einige hundert Zeilen belegte. Mit LIBRARY belegt sie nurmehr eine Zeile und wir in ca. 4 Sekunden von Disk geladen. Einmal geladen, bleibt die Library im BASIC-Code enthalten und wird mit dem Programm abgespeichert.

```
LIBRARY <Proc>,<Name>
```

bindet eine Library in den Basiccode. Die Library selbst erscheint im Programmtext nurmehr als eine Zeile in Form von *LIBRAY CODE <Proc>*. Diese Zeile stellt die nachgeladene Library dar, es werden also nicht wie mit MERGE alle Zeilen explizit dargestellt.

Beispiel:

```
LIBRARY GEM,"A:\GEM.LIB"
```

lädt die speziell gepackte GEMLIB (ab Interpreter V 3.00). Sie erscheint am Programmende in der Form:

```
LIBRARY CODE GEM
```

Leider ist es dem Anwender zur Zeit noch nicht möglich, eigene Routinen in eine LIBRARY zu verwandeln (dafür wird es demnächst einen Library-Maker geben), aber alleine für GEMLIB und die anderen OMIKRON.Libraries ist dies ungeheuer praktisch.

```
' ** MINI-VIRUSKILLER - OMIKRON **
Mem= MEMORY(512)
XBIOS Dummy,8,L Mem,L 0,0,1,0,0,1' Block lesen
WPOKE Mem,$4E75' RTS
XBIOS Dummy,9,L Mem,L 0,0,1,0,0,1' Block schreiben
```

COMPILER

Eine neue Systemvariable wurde eingeführt - COMPILER.

Wird mit Compiler gearbeitet, kann es wichtig sein, bestimmte Programmteile von der Übersetzung, bzw. vom Programmlauf auszuschließen. Dazu kann die Systemvariable Compiler benutzt werden.

```
COMPILER ohne Parameter liefert -1,
```

wenn das Programm in compilierter Form abläuft.

```
COMPILER {ON|OFF}
```

ermöglicht das Überspringen ganzer Programmteile beim Compilieren.

```
COMPILER <Stringausdruck>:
```

Im Stringausdruck werden die Compileroptionen durch Semikolon getrennt angegeben.

Beispiel:

```
COMPILER "Trace_On"
```

ersetzt die bisherige Syntax:

```
Trace_On: IF 0 then DEF PROC
Trace_On: RETURN
```

Der Interpreter, schlau wie er ist, ignoriert diesen String.

BRK

wird als ILLEGAL compiliert und löst somit im Programmlauf eine Exception über den ILLEGAL-Vektor aus. Sinn der Sache ist, daß man damit an einer bestimmten Stelle in einen im Hintergrund laufenden Debugger springen kann.

RESERVED

ist eine Systemvariable, die die Adresse des reservierten Speicherbereiches zurückgibt.

```
RESERVED(<Nummer>)
```

Die Nummer gibt den Offset an. Dokumentiert sind nur die Nummern 0 und 4.

Mit

```
PRINT LPEEK(RESVERED(0))
```

kann der Rückgabewert einer mit CALL aufgerufenen Prozedur (D0) ausgegeben werden. Es wird der letzte Wert vom Register D0 zurückgegeben.

```
POKE RESVERED(4),1
```

unterbricht INPUT USING in compilierten Programmen sofort. Allerdings muß *Multitasking_Always* aktiv sein.

Der Returnwert von INPUT USING beträgt dabei -3.

VERSION

Systemvariable, die die Versionsnummer des Interpreters liefert.

PRINT VERSION

Erscheint jetzt 300 auf dem Bildschirm, haben Sie Version 3.00 von OMIKRON.BASIC. Bitte achten Sie auf die Doppel-Null.

GEMLIB

```
FSEL_EXINPUT
```

Es ist Ihnen sicherlich schon zu Gehör gekommen, daß das TOS 1.4, das sich zur Zeit noch in Entwicklung befindet, eine erweiterte Fileselector-Box bietet, die die Übergabe einer Kommentarzeile ermöglicht. Der Anwender weiß dadurch, was in der Box gerade ausgewählt werden soll (z.B. 'Datei Laden' oder 'Speichern als'). Die GEMLIB der Version 3.00 beinhaltet diese Routine.

```
Fsel_Exinput <Pfad>,<File_Name>,-
<AusgabeString>,<Rückgabewert>
```

Ausgabestring stellt den Text dar, der oben in die Fileselektorbox geschrieben werden soll, *Rückgabewert=1* bedeutet, daß *OK* angewählt wurde, *Rückgabewert=0* steht für *ABBRUCH*. Um die Lauffähigkeit unter alten TOS-Versionen nicht einzuschränken, wird dieser Funktionsaufruf bei älteren TOS-Versionen (immerhin 99,9% der jetzigen Anwender) emuliert, was soviel heißt wie, daß zusätzlich zur gewöhnlichen Fileselektorbox eine Box mit dem Ausgabestring geöffnet wird. Beachten Sie, daß für *Fsel_Exinput* die GEMLIB eingeladen sein muß.

Viel Spaß beim Austesten.

WS/HS

Personal Computer Show 1988

Die bedeutendste Computerspielemesse Europas

Jedes Jahr im September sind die Augen und Ohren der gesamten Computerspieleindustrie auf London gerichtet. Dort treffen sich Vertreter der Presse, des Handels und der Spielehersteller zu einer Messe, die auch viele tausend Spielefans anlockt. Die Spielehersteller führen den zahlreichen Besuchern ihre neuesten Programme vor, und die Fachjournalisten laufen sich die Hacken ab, um möglichst vielen Firmen einen Besuch abstatten zu können. Ich habe mich für Sie ein bißchen auf der PCS umgesehen, die aufgrund des immensen Wachstums im Vergleich zum Vorjahr erstmalig in den Londoner Earl's Court Messehallen stattfand.

Die Messe zeigt sich trotz ausschließlich nationaler Flaggen auch international offen

Firmen wie Ocean, U.S Gold und Micropose haben keine Kosten und Mühen gescheut, um die Aufmerksamkeit der Besucher zu erregen. So hatte Ocean einen echten Panzer am riesigen Stand stehen. Die Firma Micropose, die auf Simulationen spezialisiert ist, mietete einen echten Simulator, in den Besucher einsteigen konnten, um einen simulierten Helicopterflug zu erleben. U.S Gold wartete mit einer Vielzahl von Spielautomaten und einer Reihe von hübschen Mädchen im Minirock auf, um die Massen von Besuchern an den Stand zu locken. Wegen des Riesenchaos' und des höllischen Lärms gingen viele Firmen dazu über, ihre Produkte in eigens dafür gemieteten Hotelsuiten der Presse und den Händlern zu präsentieren. So kutschierte Electronic Arts beispielsweise die Journalisten mit einer Luxuslimousine zum Hotel, in der EA-Mitarbeiter darauf warteten, die neuen Games zu demonstrieren. Lesen Sie nun, was wir in den nächsten Wochen und Monaten von den einzelnen Herstellern an neuen Spielen zu erwarten haben:

ELECTRONIC ARTS

POWERDROME ist eine futuristische Rennsimulation. Der Spieler sieht die aus dreidimensionalen Halbröhren bestehende Rennpiste aus der Cockpitperspektive. Sie können die gegnerischen Rennmaschinen zum Überholen sogar über- und unterfliegen. Nach Kollisionen in den Röhren besteht die Möglichkeit, die Rennmaschine in der Werkstatt wieder flott zu machen. *CHAINSAW WARRIOR* führt Sie in ein geheimnisvolles Haus, das Sie vollständig durchsuchen müssen. Dabei kämpfen Sie gegen Ratten, Mörder und Zombies. Außerdem kündigt Electronic Arts auch einige Umsetzungen bekannter C64-Titel an. So wird es das beliebte C64 Skateboardspiel *SKATE OR DIE* und das Autorennen *NIGEL MANSELL'S GRAND PRIX* für den ST geben.

GREMLIN GRAPHICS

Im Februar 1989 können wir *RAMRODF* erwarten. Zu Beginn hat der Spieler die Wahl zwischen zwei unwahrscheinlich starken Muskelmännern. Der eine heißt Ramrod, der andere nennt sich Rambot.

Sie laufen mit Ihrem Helden durch eine futuristische Landschaft. Ein sogenanntes "Boredom Meter" gibt an, wie langweilig das Spiel gerade ist. Während der Hatz durch die Landschaft gibt es viele Dinge zu finden, die den Helden beschäftigt halten. So können Sie ihm beispielsweise Pop-Musik vorspielen und einige andere Dinge tun, damit er sich nicht langweilt. *FEDERATION OF THE FREE TRADERS* ist eine Weltraumhandelssimulation mit 3-D Grafik, die durch Komplexität besticht.

So soll es laut Angaben von Gremlin über acht Millionen verschiedene Planeten zu besuchen geben. Das Raumschiff, das Sie steuern, ist mit verschiedenen Waffensystemen und einem Bordcomputer ausgerüstet, der mit speziellen Basicbefehlen bedient wird. *TECHNOCOP* ist ein weiteres Game von Gremlin. Es führt Sie in den rauhen Polizeialltag der Zukunft. Mit einem High Tech Sportauto rasen Sie über die Straße und zerstören die Autos der Verbrecher. Es besteht auch die Möglichkeit, Gebäude in der Stadt zu betreten und Verbrecher vor Ort mit einigen Spezialwaffen zu bekämpfen. *BUTCHER HILL* scheint ein brutales Kriegsspiel im Stil von Rambo zu sein, bei dem der Kämpfer in den vietnamesischen Dschungel geführt wird. Bei *ROY OF THE ROVERS* handelt es sich um ein neues Fußballstrategiespiel.

Fußballfreunde, die weniger Wert auf Strategie legen, sondern lieber ein heißes Fußballmatch spielen, sollten auf *GARY LINEKERS HOT SHOT* warten. Denn da gibt es Einwürfe, Eckbälle, Fouls und vieles mehr, was sich Fußballfans von einem Soccergame wünschen.

U.S GOLD

ECHELON ist der Name eines neuen Flugsimulators, in dem der Spieler eine C-104 Tomahawk im Weltraum fliegt. Per Fernbedienung können Sie ein kleines Raumfahrzeug mit eingebauter Kamera steuern., das ganz andere Perspektiven ermöglicht. Ziel des Spiels ist es, Weltraumpiraten erfolgreich zu bekämpfen. U.S Gold hat sich vorgenommen, den beliebten Spielautomaten Thunderblade umzusetzen. In diesem rasanten Game steuern Sie einen Helikopter durch die engen Straßen einer amerikanischen Großstadt und schießen auf feindliche Flugobjekte. Die *OUT RUN*-Version für den ST ist erst einige Monate auf dem Markt, da kündigt U.S Gold schon einen Nachfolgetitel an - *OUT RUN EUROPE* soll eine schnellere und um einige Features erweiterte Version des beliebten Autorennens werden. Man rast durch Landschaften, die am Straßenrand zu erkennen sind, und schlägt sich mit Ölflecken und parkenden Autos herum. *THE GAMES SUMMER EDITION* wird ebenfalls für den ST umgesetzt. Bis zu acht Spieler können ihre Fähigkeiten in acht verschiedenen Disziplinen beweisen. Da gibt es Stabhochsprung, Radrennen, Hammerwerfen, Turmspringen, Hürdenlauf, Bogenschießen und einen Gymnastikwettbewerb an den Ringen. *L.E.D STORM* ist ein rasantes Autorennen, bei dem viele Hindernisse auf der Straße liegen. Einige Lastwagenfahrer wollen Sie sogar von der Straße abdrängeln. Nach Herzenslust prügeln können Sie sich in *TIGER ROAD*. Mit Axt und stählernen Karatetritten wehren Sie sich Ihrer Heldenhaut in zahlreichen Spielabschnitten. Unter dem Label SSI veröffentlicht U.S Gold mit dem Titel *POOLS OF RADIANCE* ein neues Rollenspiel, das angeblich alle Dungeons und Städte in einer realistischen 3D-Perspektive zeigt.

OCEAN

Ocean plant ein neues Spiel mit dem Comic-Helden BATMAN. *COMBAT SCHOOL*, wo es sportliche Wettkämpfe in der Militärakademie gibt, das schon Amstrad- und C64-Besitzer begeisterte, sorgt auch bald für Krämpfe in den Fingergelenken der ST-User. Rambo kehrt zurück. Ocean bietet mit *RAMBO III* das Spiel zum Film. Das wortkarge Muskelpaket muß genau wie im Film einen Freund aus einem russischen Militärcamp in Afghanistan befreien. Daß da geschossen wird, versteht sich von selbst. In *OPERATION WOLF*, der Umsetzung des gleichnamigen Spielautomaten, geht es nicht weniger gewaltvoll zu. Sie bewegen ein Fadenkreuz über den Bildschirm und ballern auf Soldaten, die sich Ihnen in den Weg stellen. Für die BPS ein neuer Indizierungskandidat. *WEC LE MANS* ist dafür harmloser. Genau wie beim gleichnamigen Spielautomaten nehmen Sie an dem berühmt-berüchtigten 24 Stunden-Autorennen von Le Mans teil.

ELITE

GHOSTS'N GOBLINS, die süchtigmachende Spielhallenumsetzung, die schon C64- und Amstraduser in Verzücken geraten ließ, gibt es auch bald für den ATARI ST. James Bond gibt es bald auf einem Monitor in Ihrer Nähe. In einer Gemeinschaftsproduktion mit dem englischen Softwarehaus Domark gibt es ab Oktober einen neuen Bond auf Diskette. Die berühmte Motorbootjagd im Bond-Film *LIVE AND LET DIE* stand Pate für das neue Elite Game. Elite bezeichnet das Spiel als Motorbootsimulation mit vielen Actionelementen.

Na, mal sehen, was daraus wird. *MIKE READ'S COMPUTER POP QUIZ* ist ein neues Quizspiel, das sich an einer Fernsehsendung des BBC orientiert. *PAPERBOY*, schon ein Erfolg auf 8-Bit-Rechnern, kommt bald auf den ST. Ihre Aufgabe in dieser Automatenumsetzung besteht darin, die Tageszeitung in einer Kleinstadt per Fahrrad zu verteilen. Das verläuft nicht ohne Komplikationen. Hunde und rücksichtslose Autofahrer erschweren das Leben eines Zeitungsboten. *WANDERER* ist eine intergalaktische Handelssimulation mit einigen Arcadesequenzen.

PALACE

PALACE SOFTWARE kündigt *BARBARIAN II* für den ATARI ST an. Genau wie im ersten Teil kämpft der barbarische Krieger gegen den bösen Drax, um seine Geliebte Marina zu befreien. Bösewicht Drax hetzt eine ganze Meute Monster auf den barbarischen Kämpfer, die dieser mit Hilfe seiner Axt und knallharter Karatetritte beseitigen muß. Mit den *SHOOT EM UP CONSTRUCTION KIT* können ST-User ihre eigenen Ballerspiele anfertigen. Mit dem Spielebaukasten von Palace können Sie kinderleicht Grafiken und Sounds anfertigen, wichtige Features wie Kollisionsabfragen, Highscoreanzeige und Scrolling sind mit dem Shoot em up Construction Kit ebenfalls kein Problem. Um Weltraumpiraterie geht es in *COSMIC PIRATE*. In diesem Spiel müssen Sie intergalaktische Handelsschiffe ausplündern.

INFOGRAMES

In den amerikanischen Bürgerkrieg führt *NORTH AND SOUTH*. Sie planen die Strategie für den nächsten Angriff gegen die Südstaatler. Als General der siebten Company haben Sie einen schrecklichen Gegner - den Computer. Die Comichelden Tim und Struppie treiben im Comicspiel *TINTIN ON THE MOON* ihr Unwesen. Tim und Struppie wurden versehentlich in den Weltraum geschossen. Nun müssen sie sehen, wie sie in der Schwerelosigkeit zurechtkommen. *STUNTMAN*

ist ein Spiel für harte Männer. Als Stuntman haben Sie gefährliche Stunts zu bewältigen, die Kopf und Kragen kosten können. Dummerweise werden Sie bei Ihrer Arbeit auch noch von Bösewichtern verfolgt, die sich in den Kopf gesetzt haben, Sie zu töten. Na, das kann ja heiter werden!

MICROPOSE

Die Piratensimulation *PIRATES* gibt es nach Angaben von Micropose auch sehr bald für den ST. Dann können auch die ST-User durch die Weltmeere schippern, in Häfen anlegen und Handelsschiffe überfallen. Wie schwer der Job eines Fluglotsen ist, kann jeder nachfühlen, der *KENNEDY APPROACH* gespielt hat. Die einst sehr erfolgreiche C64-Simulation gibt es bald für den ST.

MIRRORSOFT

FERNANDEZ MUST DIE ist der Name eines Actionspiels, mit dem es den Spieler wieder in den Krieg zieht. Zu Fuß und per Jeep fahren Sie durch feindliches Gebiet, um Militärbasen zu zerstören. Dabei begegnen dem stählernen Kämpfer feindliche Fallschirmjäger, Flugzeuge und andere feindliche Elemente, die ihm das Heldenleben versieben. *SPEEDBALL* ist ein futuristisches Fußballspiel, bei dem man den Gegner mit einigen Waffen, die auf dem Spielfeld herumliegen, beiseite räumen kann. *BLASTEROIDS*, die getunte Asteroidsversion aus der Spielhalle, gibt es bald von Mirrorsoft für den ST. TV *SPORTS FOOTBALL* erscheint unter dem Label Cinemaware und beeindruckt durch exzellente Grafiken. Das Szenario sieht man aus der Sicht einiger Fernsehkameras.

INCENTIVE

Incentive bedient sich Freescapes, einem Entwicklungssystem, mit dem ausgefüllte und schnelle 3D-Grafiken erzeugt werden können. Auf 8 Bit-Rechnern sind die beiden Titel *DRILLER* und *DARK SIDE* schon längst ein Erfolg. Mal sehen, ob die ST-Fans auch Spaß daran haben, gefährliche Gase auf der Mondoberfläche aufzuspüren.

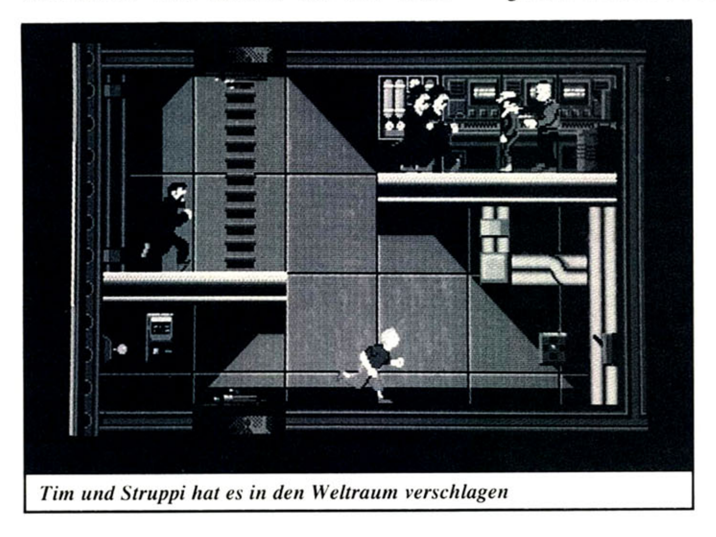
TELECOMSOMSOFT FIREBIRD / RAINBIRD

Die Adventureprogrammierer von Magnetic Scrolls (Guild of Thieves, Jinxter...) haben wieder ein neues Spitzenadventure auf der Pfanne. FISH soll es heißen. In diesem hochklassigen Adventurespiel schlüpfen Sie in die Rolle eines Fischologen, der an einem supergeheimen Projekt arbeitet: konservierendem Wasser. Abenteuerlich ist an dem Spiel, daß jemand versucht, dieses geheime Projekt zu sabotieren.

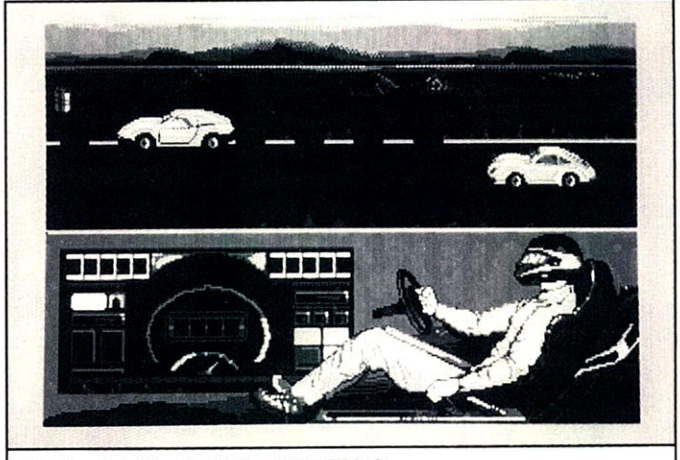
Alpträume werden in *WEIRD DREAMS* wahr. Die Hauptfigur des Spiels liegt im Krankenhaus und merkt, daß seine Alpträume zur schrecklichen Wirklichkeit werden. Plötzlich findet er sich in einer geheimnisvollen Landschaft, in der ihn gefährliche Kreaturen angreifen. Der Kampf ums Überleben beginnt. *DÉJAVU II* führt Sie nach Las Vegas. Ausgeraubt und zusammengeschlagen finden Sie sich in einem miesen Hotel wieder. Als Detektiv müssen Sie die Umstände des Überfalls klären. Genau wie im Vorgängeradventure sind Texteingaben überflüssig. Alle Eingaben werden mit der Maus gesteuert. ELITE, die beliebte Handelssimulation im Weltraum, die auf allen Systemen schon absoluter Spitzenreiter in den Top Ten war, wird bald für den ST erhältlich sein. Im Wilden Westen geht es hoch her. Das wird an *BLAZING BARRELS* deutlich. Da wird geschossen, was das Zeug hält. Den Sheriff und den Deputy hat es schon erwischt. Nun liegt es an Ihnen, alle Banditos in die ewigen Jagdgründe zu schicken.

HEWSON CONSULTANTS

Eines der originellsten 8 Bit-Spiele im letzten Jahr war *NEBULUS*. Jetzt können auch die ST-User rotierende Türme erklimmen und jeweils im obersten Stockwerk eine Bombe aktivieren. *ZYNAPS* ist ein Ballerspiel mit horizontalem Scrolling, vielen Extrawaffen und noch mehr Bildschirmgegnern. Beide Spiele müßten zum Zeitpunkt des Erscheinens dieses Artikels schon veröffentlicht sein.



Tim und Struppi hat es in den Weltraum verschlagen



Ein Joystick für alle Fälle - STUNTMAN

MASTERTRONIC

In Revenge II läuft man bei horizontalem Scrolling mit einem Kamel über eine Planetenlandschaft und schießt auf skurrile Gestalten. Das Game hat mich schon auf dem C64 nicht überzeugt. *MOTORBIKE MADNESS* scheint dagegen schon interessanter zu sein. Knifflige Hinderniskurse werden den Spieler lange Zeit in Atem halten.

MELBOURNE HOUSE

Ein hübsches Mädchen wurde vom bösen Big Boss Willy, dem Kopf einer gemeingefährlichen Gangsterbande, gekidnapt. Ihre Aufgabe im Spiel *DOUBLE DRAGON* besteht darin, sich durch die Straßen zu kämpfen und die Straßenbande zu besiegen. Dabei stehen viele Waffen zur Verfügung.

CBO

Der Kommentar

VON CLAUS PETER LIPPERT

London im Nebel

Eindrücke einer Computer-Messe

Verloren in einem verwirrenden Labyrinth von Gängen und Ständen, benebelt vom Klebstoffgestank frisch errichteter Stände und den Nachwirkungen abendlicher Pressepartys, dies kann nur die Personal Computer Show in London sein - und sie war es auch.

Panem et circensis - nach der Abfütterung der anwesenden Journalisten durch diverse Partys in den besten Londoner Hotels, waren Spiele angesagt.
Wie immer waren sie leicht zu finden: Die Stände der "Unterhaltungssoftwareanbieter" waren ungefähr so bevölkert wie Mallorca im Juli, was bei zahlreichen der repräsentierenden Damen wohl zu Mißverständnissen auf der Bekleidungsebene führte.
Die Aussteller hatten wie immer eine klare Vorstellung von ihrem Publikum und wie man es mit aufwendiger Technik, hervorragender Präsentation und der lautesten Musik betäubt. Die PCW ist bekannt für verrückte Stunts, als Kleidung fehlinterpretierte Verpackungskunst und das Herumschütteln unvorsichtiger Besucher in Hubschraubersimulatoren. Wenn Sie sich für "Action", Gedränge und Herzstillstand interessieren, sind Sie bei der PCW genau richtig.

Jahr für Jahr bauen die Hauptakteure in diesem großen Spiel riesige Stände, um sich und den Betrachtern zu beweisen, daß es ihnen besser geht als allen anderen. Das Resultat ist ein beeindruckendes Spektakel, das den Besucher richtiggehend anspringt, kaum daß er die Halle betritt. Benebelt von der Geräuschkulisse und den strengen Düften frischen Klebstoffs - die Standbauer sind erst morgens um fünf fertig geworden - wankt er dann dahin, ein willfähriges Opfer elektronischer Illusionen.

Reizende Aussichten

Natürlich gibt es auch noch andere Attraktionen bei der PCW. Auf der mehr menschlichen Ebene ist diese Ausstellung überfrachtet mit entzückenden Mädels, die alle Arten von Dingen offerieren - werfen wir also einen Blick auf dieses reizende Angebot:

Neben dem Herumgeschütteltwerden im Simulator von Mikroprose (14 Sitze, inklusive der Garantie, es sei wirklich "scary") und dem Angebot von Epyx ein "echtes Kawasaki Motorrad" zu gewinnen, gibt es stets Hunderte von kostenlosen Buttons (originelle Inschriften wie: I love f(x) mit x=jede beliebige Softwarefirma), T-Shirts (mit jenen originellen Inschriften) und jede Menge Poster - von den neuen Spielen oder einfach dem Firmenlogo von f(x) (...).

Natürlich kann man diese Show nicht besuchen, ohne so ungefähr jede Spielidee zu sehen, die jemals für acht, sechzehn oder zweiunddreißig Bit entwickelt wurde - und dies gleich mehrmals. Das beliebteste Spiel war zweifellos "Show Special". Wo es angeboten wurde, standen die Menschen Schlange - und es wurde überall angeboten. In der Tat kann man bei einer Messe so viel sparen, daß man davon locker seinen Lebensunterhalt bestreiten könnte.

Brot...

Während das findige Reporterauge nach Attraktionen Ausschau hält, kriecht (ich verlasse gerade eine Präsentation, da tritt mir so ein besoffenes Schw... auf die Hände) und schiebt sich sein Besitzer in der Menschenmasse vorwärts zur "Pepsi Challenge". Dort gilt es den Geschmackstest zu bestehen - aus neutralen Bechern wird Pepsi und Coke geschlürft (Geschmacksrichtung: Plastik, Jahrgang 88). Irgendwann vor Weihnachten soll dann das Ergebnis (in Zusammenarbeit mit US Gold) der staunenden Öffentlichkeit präsentiert werden. Erste Hochrechnungen ergaben dann auch, daß fast 61% aller Computerabhängigen Pepsi vergleichbaren Substanzen wie Korrosionsschutzlösung vorziehen. Daß dieses Urteil davon beeinflußt sein könnte, daß es für die richtigen Antworten "...fabulous free prizes..." gibt, ist kaum vorstellbar.

...und Spiele

Ein weiterer Hauptakteur der Szene ist Ocean. Dort hat man bereits in der Vergangenheit so erfolgreiche Antikriegsspiele wie Platoon (als einziges Programm bereits vor seinem Erscheinen in Deutschland indiziert) veröffentlicht. Die Neuankündigungen setzen diesen gewinnträchtigen Trend fort. Demnächst erscheint Rambo III auf den heimischen Monitoren (eine herzzerreißende Satire auf intellekt- und niveaulose Gewaltfilme wie "Good morning Vietnam"). Aber Vorsicht, Ocean! Grandslam, berüchtigt durch "Hunt for Red October", hat bereits angekündigt, durch weitere Produkte aktiv an der Befriedung des Bildschirms teilzunehmen. Sie werden leider nicht die Einzigen bleiben.

Und wer schon immer einmal so richtig für einen Augenblick berühmt sein wollte und auch diesmal wieder nicht in der KICKSTART abgebildet ist, der hat in London gefehlt. Dabei sagt doch schon Lafunze, der alte chinesische Philantroph: "...dabei sein ist alles..." (wie sollte man sonst auch an all die kostenlosen Poster, Buttons, T-Shirts, Magazine und Drinks kommen). Wenn Ihnen also diese "close encounter of the special kind" noch fehlt, auch nächstes Jahr wird über Earls Court wieder das jüngste Gericht hereinbrechen - schauen Sie doch mal rein. ■

fibuMAN 3.0

Das Bessere ist der Feind des Guten

Das von uns bereits im Heft 12/87 getestete Finanzbuchhaltungsprogramm fibuMAN ist jetzt in der erweiterten Version fibuMAN 3.0 erschienen. Da fibuMAN mit zu dem Besten gehört, was an Programmen für den ATARI angeboten wird, war ich natürlich gespannt, was dem Autor noch an Verbesserungen eingefallen war. Um es gleich vorwegzunehmen: Das Programm wurde nicht nur an die neuesten gesetzlichen Vorschriften nach dem Bilanzrichtliniengesetz angepaßt, es hat auch noch eine ganze Reihe von wesentlichen Verbesserungen erhalten.

Der Anwenderkreis

Die Einnahme-Überschuß-Rechnung *fibuMAN e* ist gedacht für alle, die laut Paragraph 141 AO nicht zur Bilanzierung gezwungen sind, und nur alle Geschäftseinnahmen den -ausgaben gegenüberstellen müssen, wie z.B. Freiberufler.

Für Geschäftsleute, die laut Gesetzgebung zum Führen von Büchern gezwungen sind, und für kleine Kapitalgesellschaften, die sich dem neuen Bilanzrichtliniengesetz anpassen müssen, ist *fibuMAN f* gedacht.

Soll die Buchführung mehrerer Firmen mit fibuMAN erledigt werden, ist *fibuMAN m*, die mandantenfähige Finanzbuchhaltung, angesagt. Mit dieser Version des Programmes kann sowohl eine Finanzbuchhaltung als auch eine Einnahme-Überschuß-Rechnung gefahren werden. Besonders interessant ist dieses Faktum für diejenigen Steuerpflichtigen, die gerade noch eine Einnahme-Überschuß-Rechnung machen können, aber auf der Schwelle zur Bilanzierungspflicht stehen. Sie können durch einfaches Umschalten in der Option Firmenparameter ihre Daten sowohl als Einnahme-Überschuß-Rechnung als auch als Bilanz und Gewinn- und Verlustrechnung auswerten. Integriert in *fibuMAN m* ist auch die betriebswirtschaftliche Auswertung, die bei *e* und *f* noch zusätzlich zu erwerben ist.

Kompatibilität

fibuMAN ist ab Version 3.0 im Sinne einer noch größeren Flexibilität mit dBASE-kompatiblen Datenbankdateien ausgerüstet. Dennoch brauchen die Anwender älterer Versionen nicht zu fürchten, daß Ihre Datenbestände nicht mehr weiterverwandt werden können; jedem Update wird ein Konvertierungsprogramm beigefügt, das die Umwandlung auf Knopfdruck vornimmt.

Hier in kurzer Form einige wesentliche Unterschiede der beiden fibuMAN Versionen:

Geschwindigkeit

fibuMAN in der Version 3.0 wurde im schnellen dBMAN 5.0 mit dem Compiler "Greased Lightning" von Versasoft geschrieben und hält, was der Name des Compilers verspricht: fibuMAN geht ab wie ein "geölter Blitz". War schon die Schnelligkeit in der zuerst getesteten Version beeindruckend, so übertrifft das neue Blitz-fibuMAN selbst die größten Erwartungen. Durch Einsetzen der Cache-Memory war es mit dem Compiler möglich, die Geschwindigkeit beim Buchen, bei Auswertungen und Listen je nach Programmpunkt ums bis zu Zehnfache zu steigern.

Das Handbuch

Völlig überarbeitet wurde das fibuMAN Handbuch. Die Autoren lernten aus Anwenderanfragen. Das, was in der Hotline am meisten abgefragt wurde, kam als neues oder erweitertes Kapitel zum Handbuch dazu. Geblieben ist die Gliederung des Handbuchs, das als Nachschlagewerk dienen sollte. Integriert wurde die überarbeitete Checkliste (Was darf man? Was sollte man möglichst nicht tun? Was muß unbedingt beachtet werden?) und das Kapitel "Die erste Buchung". Die fibuMAN-Autoren haben gerade in diesem Kapitel aus einem Manko des alten Handbuchs gelernt und zusätzlich noch eine kurze Einführung in die Technik der doppelten Buchführung mit diversen Probebuchungen gegeben.

Datensicherheit

In puncto Datensicherheit, einem sehr wichtigen Punkt bei einem Buchhaltungsprogramm, ließ fibuMAN auch in der zuvor getesteten Version 2.0 schon keine Wünsche mehr offen. Sogar bei schwerwiegenden Bedienungsfehlern wie Ausschalten des Rechners während des Programms und somit nicht geschlossenen Dateien - fibuMAN ist eine Datenbankapplikation - konnte mit dem beigefügten "Retter"-Programm die Buchführung wiederhergestellt werden. FibuMAN 3.0 geht jetzt noch einen Schritt weiter. Wurden durch Stromausfall, defekte Hardware oder gröbste Bedienungsfehler wie Aussschalten des Rechners Dateien nicht ordnungsgemäß geschlossen, wird beim erneuten Starten des Programms eine automatische Datenregeneration aufgerufen. Zusätzlich erfolgt jetzt noch nach jeder 50sten Buchung und beim Verlassen eines jeden Eingabeteils eine Zwischenspeicherung aller Daten, die gewährleistet, daß bei einem Stromausfall, Reset o.ä. noch nicht einmal die automatische Datenregeneration in Anspruch genommen werden muß, sondern sofort mit intakten Dateien weitergearbeitet werden kann.

Darüberhinaus legt fibuMAN 3.0 bei jedem Monatsabschluß automatisch eine Kopie des Monatsjournals als zusätzliche

Sicherheitskopie auf dem Datenpfad ab, sofern mit der Festplatte gearbeitet wird.

Bedienerführung

Die Bedienerführung wurde noch weiter ausgebaut. Ein Beispiel: In der Praxis hatte sich z.B. erwiesen, daß es sinnvoll ist, die Kontoauszüge nicht nur jederzeit komplett für das gesamte Jahr, sondern auch monatlich ausdrucken zu können. Dementsprechend wurde der Programmteil “Kontoauszüge” erweitert. Ebenso wurden die verschiedensten Optionen durch den Menüpunkt “Ausgabe auf : Drucker, Diskette, Bildschirm” zur weiteren Komfortsteigerung erweitert.

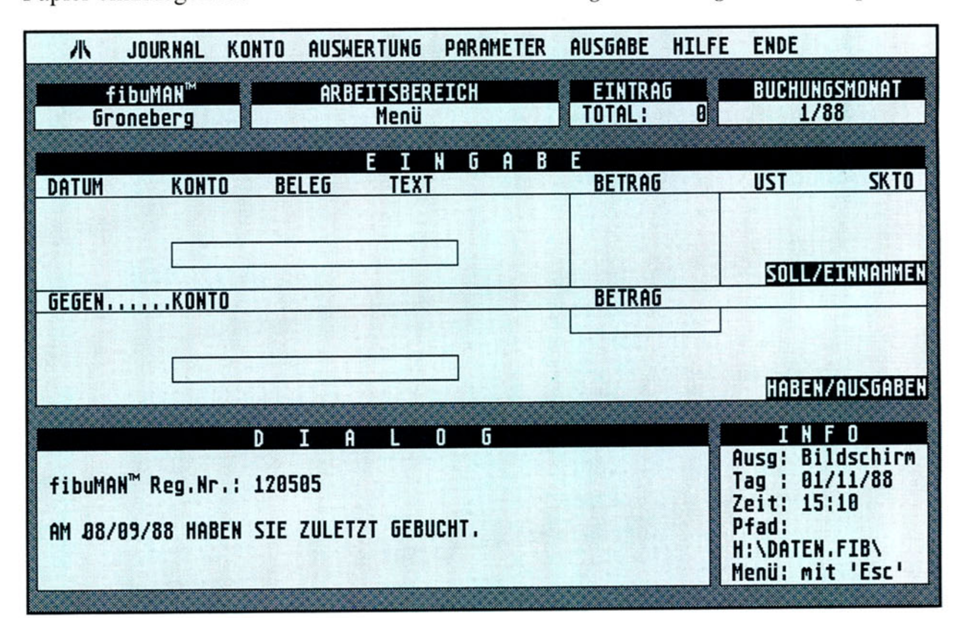
Erweiterte Auswertungen

Obwohl schon in der Version 2.0 der Menüpunkt “Auswertung” das Prunkstück war, wurden hier noch weitere Möglichkeiten geschaffen:

Entsprechend dem Bilanzrichtliniengesetz ist das Bilanzschema noch übersichtlicher geworden. Neben dem Ausweis von ausstehenden Einlagen, wird in der neuen Version auch der nicht durch Eigenkapital gedeckte Fehlbetrag auf der Aktivseite angezeigt. Des weiteren ist die saldenabhängige Aktivierung und Passivierung von Forderungen und Verbindlichkeiten hinzugekommen. Ein Umsatzsteuerguthaben erscheint im neuen Bilanzschema nicht mehr auf der Passivseite mit negativem Vorzeichen, sondern, wie der Gesetzgeber vorsieht, positiv auf der Aktivseite als Forderung. Negative Banksaldi werden nicht mehr mit den übrigen liquiden Mitteln aktivisch verrechnet, sondern stehen als kurzfristige Forderungen auf der Passivseite. Der Autor hat noch weiter gedacht. Vergaßen Anwender, im Rahmen der Eröffnungsbilanz den Saldovortrag des Kapitalkontos zu buchen, zeigte sich logischerweise eine Differenz zwischen Bilanz und Gewinn- und Verlustrechnung bezüglich des Jahresüberschusses bzw. -fehlbetrags. In der neuen Version wird der Differenzbetrag automatisch in der Bilanz unter Eigenkapital als Wert für die Eröffnungsbuchung aufgeführt.

Umsatzsteuervoranmeldung, Umsatzsteuererklärung

Neben dem neuen Ausweis der umsatzsteuerfreien Umsätze in allen Versionen bietet fibuMAN 3.0 jetzt die Möglichkeit, im Rahmen der Umsatzsteuer das Berlin-Förderungsgesetz zu berücksichtigen. Ebenso verbessert wurde die Anpassung des Drucks der Umsatzsteuervoranmeldung an die amtlichen Formulare, die, man sollte es nicht glauben, innerhalb der Bundesrepublik unterschiedlich ausfallen. Zusätzlich wurde der Formulardruck erleichtert, indem vor dem eigentlichen Druckvorgang ein Probedruck stattfinden kann, der millimetergenau angibt, wie das Papier einzulegen ist.



Abschreibungen

Es ist jetzt möglich, Abschreibungsprozentsätze bis 99.99% einzugeben und somit auch gebrauchte, aber neu gekaufte Anlagegüter mit einer Abschreibung von 1 - 2 Jahren einzugeben (Version 2.0 nur bis 35%). Darüberhinaus werden, entsprechend dem Bilanzrichtliniengesetz, auch die kumulierten Abschreibungsbeträge angezeigt (Bruttoprinzip).

Veränderungen 2.0 -> 3.0 während des Buchungsvorgangs: Völlig neu in fibuMAN 3.0 integriert wurde die automatische Skontoberechnung und Buchung. Bei Buchung des Zahlungsein- oder -ausgangs (und nur da, sonst erfolgt ein entsprechender Hinweis) kann ein Skontoprozentsatz eingegeben werden, der sofort den entsprechenden Skontobetrag ausrechnet. Dieser Betrag kann natürlich überschrieben werden, so daß auch die Möglichkeit besteht, einen Skontobetrag zu buchen, wenn der Prozentsatz nicht bekannt ist. Die anteilige Mehrwertsteuer wird gegebenenfalls automatisch zu- bzw. abgerechnet. fibuMAN bucht anschließend einmal den Zahlungsein-/ausgang, dann den eingegebenen Skontobetrag.

Kassenbuch: fibuMAN enthält ein vollständiges Kassenbuch, das während der Buchung automatisch und ohne Zusatzeingaben mitgeführt wird.

Uhr- und Ordnerprogramm: Das Programm zum Stellen der Systemzeit wurde analog zu den übrigen Eingabemodi von fibuMAN abgestimmt.

Unsinnige Buchung: Ein Knopfdruck genügt neuerdings, um Soll- und Haben-Fehler, auf die das Programm schon immer selbsttätig aufmerksam machte, auszumerzen.

Klingeln bei leerer Kasse: Sollte man versuchen, eine Rechnung aus einer Kasse zu begleichen, in der hierfür nichts oder nicht genug ist, klingelt fibuMAN neuerdings unüberhörbar, sofern dies in den Firmenparametern gewünscht wurde. (Ich wünsche jedem Anwender, daß es nicht zu oft klingelt!)

Schnittstellen: dBMAN ab 4.0, StandardBase, dBASE, ASCII-Schnittstelle möglich.

Analyse: In Zusammenarbeit mit dem schon lieferbaren *fibuSTAT* ist eine grafische Analyse zur Betriebsauswertung über einen Zeitraum von 5 Jahren möglich.

Blättern in Listen/Suchen nach Datensätzen: Neues Bildschirmlayout mit Überschriften in allen Bildschirmlisten mit der Option der Datensatzauswahl mittels durch die Cursortasten zu steuerndem invertierten Balken. Hiermit ist z.B. die

Auswahl von Konten nicht mehr nur durch die Kontonummer, sondern durch einfaches Daraufzeigen im Kontenplan möglich. Dies entlastet das Gedächtnis enorm, so daß man sich voll auf die eigentlichen Buchungen konzentrieren kann.

Zusammenfassung

Die Auflistung der wichtigsten Änderungen läßt schon erkennen, wie umfassend das Programm erweitert wurde. Ganz offensichtlich ist dabei, daß die Fragen und Anregungen aus der "Hotline" Grundlage der Verbesserungen waren. Die meisten Änderungen dienen, abgesehen von den Anpassungen an das neue Bilanzrichtliniengesetz, der verbesserten Handhabung, dem Komfort und der Buchungssicherheit. Erfreulich dabei ist auch, daß trotz der umfassenden Neugestaltung des Programmes der alte Preis gehalten wurde.

Das einzige, was eventuell stören könnte, ist die Begrenzung auf nur einen Kontenrahmen (DATEV SKR 03). Es soll jedoch noch in 1988 eine Version mit frei wählbarem Kontenrahmen kommen. Ansonsten ist das Programm für fast jeden Anwendungsbereich im Bereich der Finanzbuchhaltung absolut zu empfehlen.

Michael Groneberg

Preisübersicht:

fibuMAN DEMO	DM 65,-
fibuMAN e	DM 398,-
fibuMAN f	DM 768,-
fibuMAN m	DM 968,-
BWA zu fibuMAN f	DM 98,-
fibuSTAT	DM 398,-
faktuMAN Komplettpaket	DM 845,-
faktuMAN Einzelmodule ab	DM 95,-
Import-Modul	DM 148,-
Buchführungskurs	DM 345,-

Noch ein Hinweis für die Benutzer der älteren Version und der Hotline: Mit dem Umzug hat sich nicht nur die Anschrift, sondern auch der Firmenname geändert. Die damalige Firma PRODATA, Bergisch Gladbach, heißt nun

NovoPLAN SOFTWARE GmbH
mit dem Firmensitz in
Hardtstraße 21
4784 Rüthen 3
Tel. 02952/8080

ENDE

Turbo C contra Laser C

Nachdem wir Ihnen die Vorabversionen der beiden Entwicklungssysteme bereits vorgestellt haben, präsentieren wir Ihnen hier noch einmal die endgültigen Produkte. Ring frei zur zweiten Runde...

Turbo C aus dem Hause Heimsoeth/Borland steigt mit einem Kampfgewicht von 660 deutsch geschriebenen Handbuchseiten und drei einseitig formatierten Disketten in den Ring. Der Kontrahent Laser C wartet mit 640 englischen Seiten (das deutsche Handbuch ist noch in Arbeit) und zwei doppelseitigen Floppies auf. Lassen wir die beiden den Ring (Partition D meiner Harddisk) besteigen, um sich zu messen.

Turbo C gibt sich unkompliziert. Neuen Ordner anlegen, die drei Floppies hineinkopieren, *TC.PRG* anklicken, läuft. Die Installation auf Disketten gestaltet sich ähnlich einfach. Ist man mit einem einseitigen Floppylaufwerk gestraft, muß man auf die Hilfstexte von Turbo C vezichten. Kein Wunder, denn die Datei mit den Hilfstexten ist fast 300 KB groß.

Vom Prinzip her geht das alles bei Laser C genauso einfach. Dummerweise merkt die Laser C-Shell nicht, von welchem logischen Laufwerk sie gestartet wurde, und tut dementsprechend auch nicht das Naheliegende, nämlich alle anderen Dateien der Entwicklungsumgebung auch von diesem Laufwerk laden. Nein, beim ersten Start von der Harddisk lädt die Shell alle ihre benötigten Daten zunächst von Laufwerk A. Dann müssen die Unterverzeichnisse für die Hilfsprogramme,



Bibliotheken und so weiter auf Laufwerk D (in meinem Fall) gesetzt und in der Konfigurationsdatei gespeichert werden. Erst dann kommt auch mit der Harddisk Freude auf. Wer sich immer noch nicht an grafische Benutzeroberflächen gewöhnt hat und ein Feind des kleinen grauen Tierchens ist, kann auch einen der vielen Kommandointerpreter, die es für den ST inzwischen gibt, benutzen, um seine Programme zu erstellen. Beide Systeme sehen diese Möglichkeit vor. Bei Laser C kann man sogar die Shell dazu verwenden, in einem Fenster über Kommandos mit Compiler, Linker und was es sonst noch so gibt zu kommunizieren.

Für beide Systeme benötigt man einen ATARI ST mit mindestens einem Megabyte Speicher und TOS im ROM. Will

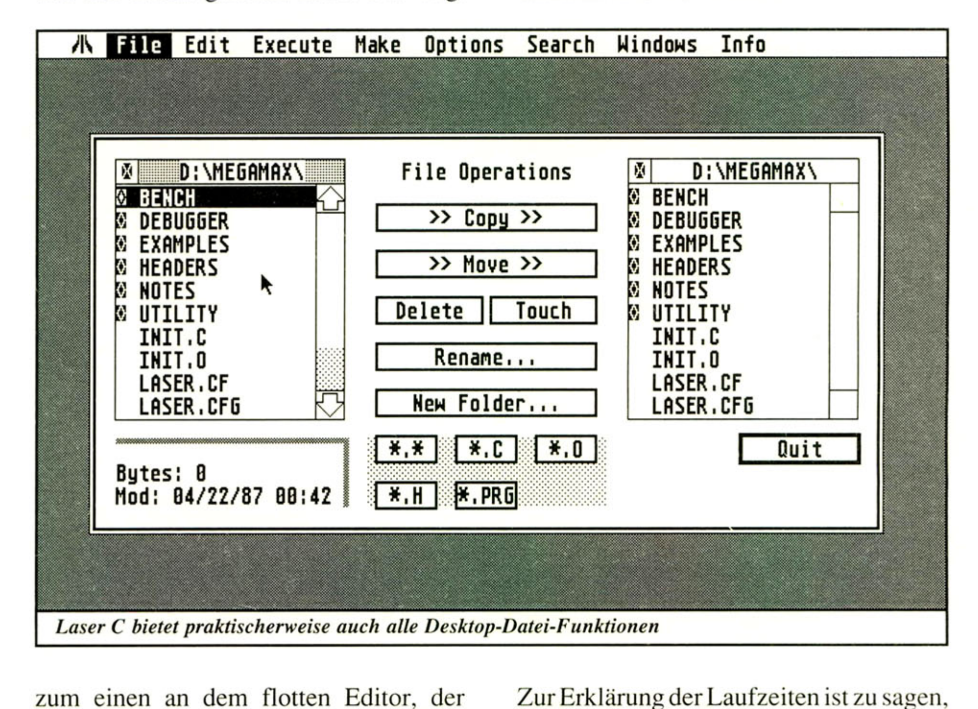
man Laser C mit einem einseitigen Diskettenlaufwerk betreiben, muß man jemanden haben, der die doppelseitigen Originaldisketten auf einseitige umkopiert. Fraglich bleibt dabei, wer mit so einem System überhaupt noch arbeitet, geschweige denn programmiert. Trotzdem finde ich es erfreulich, daß die Entwickler dies berücksichtigt haben. An dieser Stelle möchte ich auch auf einen Mangel hinweisen, auf den mangelnden Kopierschutz der beiden Kandidaten. Meiner Meinung nach sollte jedes Handbuch den Satz "Bitte fertigen Sie zuerst Sicherheitskopien Ihrer Originaldisketten an und verwahren Sie die Originale an einem sicheren Ort" enthalten. Die Handbücher von Turbo und Laser C tun es. Und sie enthalten noch eine ganze Menge mehr (kein Wunder bei dem Umfang). Mehr als die Hälfte verschlingt die Beschreibung der Bibliotheken, als da wären AES, VDI, BIOS, XBIOS, GEMDOS, UNIX-kompatible Funktionen und die Line-A- Bibliothek. Letztere braucht im Turbo C-Handbuch nicht beschrieben zu werden, weil sie dort nämlich fehlt. Mit 21 Seiten ist der Index der Turbo C-Dokumentation dreimal so lang wie der von Laser C. Beide Handbücher sind übersichtlich gegliedert und relativ vollständig. Besonders gut haben mir bei Turbo die Erläuterungen zur Implementation des Compilers gefallen.

Shell und Editor

Beide Systeme werden über eine GEM-Shell bedient und sind einfach und schnell zu handhaben. Beide stellen Tastaturkürzel für die Menüfunktionen zur Verfügung, ein allgemeiner und, wie ich meine, vorteilhafter Trend. Der Teufel steckt natürlich, wie immer, im Detail, und welche Benutzerschnittstelle ist schon so perfekt, daß niemand mehr eine Verbesserung einfällt? Tatsächlich ist es so, daß beide Systeme sowohl Stärken als auch Schwächen haben und eine Kombination aus beiden ideal wäre. Das gilt nicht nur für Shell und Editor, sondern allgemein. Im folgenden möchte ich eine Liste der Vor- und Nachteile aufführen. Beginnen wir mit Turbo. Die Tastaturkommandos des Editors sind stark an die des Apple Macintosh angelehnt. Da ich öfter mit dem Mac als mit dem ST arbeite, bin ich davon natürlich begeistert. Allerdings bin ich Mac II-Geschwindigkeit gewohnt, und der Editor von Turbo ist ziemlich lahm. Beim Scrollen läßt er sich ca. fünfmal soviel Zeit wie der Laser C- Editor, der seinerseits in dieser Disziplin an *Tempus* herankommt. Toll finde ich die Möglichkeit, durch einen Doppelklick auf die Zeile, in der eine Fehlermeldung des Compilers steht, auf die dazugehörige Zeile im Quelltext zu springen. Für diesen Fall bietet Laser C keine komfortable Lösung an. Sehr bemerkenswert ist die Hilfsfunktion von Turbo C. Sie ersetzt in den meisten Fällen das Handbuch. Vergleicht man die Hilfstexte mit der Dokumentation, entdeckt man auffällig große Ähnlichkeit. Auch der Umfang ist mit ca. 300 KB (in Worten dreihundert Kilobyte) so groß, daß die Bezeichnung On-Line Manual (ich weiß beim besten Willen keine deutsche Übersetzung für diese Redewendung) eigentlich treffender wäre.

Nun zu Laser C: Was die turn-around Zeiten angeht, ist Laser C das Schnellste, was ich bisher gesehen habe. Das liegt zum einen an dem flotten Editor, der wirklich in vielen Funktionen an *Tempus* heranreicht und zum anderen an dem *Cache*-Prinzip der Shell. Das mit dem Cache funktioniert so: Programme (Compiler, Linker, Make usw.) und Daten werden im Hauptspeicher gehalten, und man spart den Zugriff auf einen (langsamen) Massenspeicher. Man darf das aber nicht mit einer RAM-Disk verwechseln. Bei der muß nämlich das auszuführende Programm trotzdem noch in den Speicher geladen bzw. umkopiert werden. Das kostet zusätzlichen Platz und natürlich auch Zeit. Bei Laser C wird z.B der Compiler nicht mehr im Speicher umkopiert, sondern direkt gestartet, und das geht erheblich schneller als das Laden von einer RAM-Disk. Die genauen Zeiten stehen im Kasten nebenan. Die Zahlen neben den Programmnamen geben die Länge des Quelltextes in Bytes an. Die untere Zahl bei Laser C (2.Mal) gibt die Zeiten an, die gemessen wurden, wenn der Quelltext bereits im Cache der Shell war. Ein weiteres Plus für die Laser-Shell ist der integrierte Kommandozeileninterpreter für die UNIX-Süchtigen. Auch die mitgelieferten Hilfsprogramme können sich sehen lassen. Das Make ist eine der wichtigsten Anleihen an UNIX und übertrifft die Projektverwaltung von Turbo bei weitem. Soviel zu Shell und Editor.

Der Compiler Turbo erzeugt schnelleren und kürzeren Code. Die exakten Werte entnehmen Sie bitte den beiden Tabellen.



Laser C bietet praktischerweise auch alle Desktop-Datei-Funktionen

Zur Erklärung der Laufzeiten ist zu sagen, daß es bei dem ersten Wert jeweils Optimierungen, Registervariablen usw. ausgeschaltet und beim zweiten Wert eingeschaltet waren. Die Ergebnisse für *Sieve* sind in Sekunden angegeben, die *Dhrystones* sind in Dhrystones pro Sekunde gemessen. Die Ergebnisse des Dhrystone-Tests sind ungewöhnlich hoch. Das liegt vermutlich an der cleveren Optimierung von Turbo C. Ein Kommentar im Quelltext warnt vor den Ergebnissen eines optimierenden Compilers, da der Test jede Menge Operationen enthält, die wegoptimiert werden können. Zum

	DHRY.C 36955 TIME.C 258	APPACC.C 1184
Laser C 2. mal	15,9 s 14,5 s	6,1 s 5,6 s
Turbo C	29,1 s	20,2 s

Bild 1: Zeiten zum Compilieren und Linken

Beispiel die Zuweisung eines Werts an eine Variable, die anschließend nicht mehr verwendet wird. Anders kann ich mir das Ergebnis auch nicht erklären. Der Macintosh II, der einen 68020 mit 16 MHz Takt verwendet, kommt mit dem schnellsten mir bekannten Compiler (MPW 2.02) gerade auf ca. 2500 Dhrystones. Ich glaube, die Geschwindigkeiten des Siebs sind aussagekräftiger, und das bedeutet, daß Turbo ca. 25% schneller ist als Laser C. Das ist zwar nicht mehr so

	LEER	APPACC	DHRYSTONE
Laser C	3350	4403	14046
Turbo C	410	1231	5480

Bild 2: Programmlängen in Byte

überwältigend wie die ca. 100%, die der Dhrystone-Test suggeriert, aber immer noch beeindruckend.

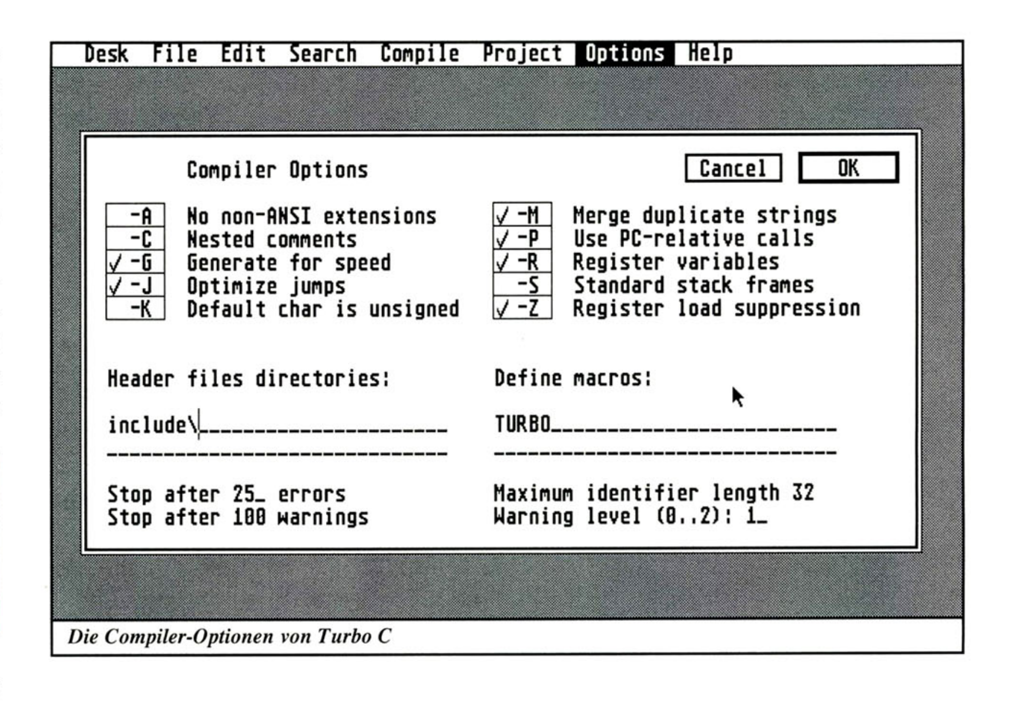
Haben Sie schon vom *ANSI-C-Standard* gehört? Den erfüllt nämlich Turbo. Was verbirgt sich dahinter? Erweiterungen zum K&R-Standard, die von der amerikanischen Normungsbehörde *ANSI* gerade definiert werden. Einige davon sind auch in Laser C vorhanden, als da wären Strukturzuweisungen, Aufzählungstypen und der Typ void für Funktionen, die keinen Wert zurückliefern. Der wesentliche Vorteil von ANSI-C sind aber die Funktionsprototypen, die es erlauben, die Typen der Übergabeparameter an Funktionen anzugeben. Der Compiler prüft dann beim Aufruf der Funktion, ob die Parameter zulässig sind. Außerdem stellt ANSI-C zusätzliche Bibliotheksfunktionen zur Verfügung. Zum Beispiel *vprintf*. Die Wirkung ist identisch mit dem be-

	Sieve	Dhrystones
Laser C	4,22 2,52	816 848
Turbo C	2,00 2,00	1439 1708

Bild 3: Programmlaufzeiten

kannten *printf*, lediglich die Übergabe der Parameter ist unterschiedlich. Während printf eine Parameterliste als Argument benutzt, verwendet *vprintf* einen Zeiger auf eine Parameterliste.

Eine C-Erweiterung, die nur Laser C anbietet, ist die Möglichkeit, Assembler direkt in C-Programme einzubinden. Bei Turbo C müssen Assembler und Debugger zusätzlich erworben werden. Da beide zum Zeitpunkt dieses Tests noch nicht



Die Compiler-Optionen von Turbo C

fertig waren, kann ich dazu nur soviel sagen, daß der Assembler 680X0 (CPU), 68881-68882 (FPU) und 68851 (MMU) Code erzeugen kann. Der Debugger arbeitet auf Assemblerebene und versteht die Symbole, die der Compiler erzeugt.

Bleibt zu den beiden Compilern nur noch zu sagen, daß Turbo C Objektdateien im DRI-Format erzeugt, und Laser C ein eigenes Objektformat verwendet. Allerdings verstehen Linker und Librarian von Laser C auch DRI-Objektdateien.

Das Ambiente

Allgemein ist es so, daß Laser C eine Menge Hilfsprogramme enthält, die für die Entwicklung nützlich sind. Turbo C ist in dieser Hinsicht eher spartanisch ausgestattet. Laser C enthält zusätzlich zu Shell, Compiler, Linker und Librarian ein

```
#ifdef LASER
#include <gemdefs.h>
#endif

#ifdef TURBO
#include <AES.H>
#endif

extern int _app;

main_loop()
{
    form_alert(1,"[1][Hello World][OK]");
}

main()                              {
   int desk_id,
       appl_id,
       msg_buf[8];
      appl_id = appl_init();          /* Hallo AES hier bin ich          */
      if(_app) {                      /* PRG oder ACC ?                  */
      main_loop();                    /* Programm ausführen              */
   }
   else {                             /* Titel eintragen in's Deskmenü   */
     if((desk_id = menu_register(appl_id, "  Hello World")) == -1)
        {                             /* Schade, schon voll, sorry user */
        form_alert(1,"[1][Kein Platz mehr in der Menueleiste][ OK ]");
        appl_exit();                                 }
     do {                             /* warten auf eine Nachricht...    */
        evnt_mesag(msg_buf);
        if((msg_buf[0] == AC_OPEN) && (msg_buf[4] == desk_id))
           {
           main_loop();               /* Wir sind gemeint, also Los      */
           }
        } while(1);                   /* immer und immer wieder...       */
}
   appl_exit();                       /* bis zum bitteren Ende           */
}
```

Listing: So kann man eine Anwendung wahlweise als Accessory oder normales Programm starten.

Resource Construction Set, einen Debugger, ein Hilfsprogramm zum Untersuchen von Textdateien (*egrep*), ein Programm zum Disassemblieren von Objektdateien und einige Hilfsprogramme aus der UNIX-Toolbox (*ls*, *cat*, *echo* usw). Das RCS liegt in der Version 2.0 vor. Gegenüber älteren Versionen können Resourcen bis zu 64 KB Länge (früher 32 KB) erzeugt werden. Ein einzelner Objektbaum kann jetzt maximal 2000 Objekte enthalten (früher 256). Außerdem kann eine Resource auch als C-Quelltext abgespeichert werden. Um diesen Quelltext auch praktisch nutzen zu können, wird in den Bibliotheken eine Funktion bereitgestellt, die den Objektbaum dem GEM bekannt macht. Dabei wurde auch an Entwickler gedacht, die ein anderes Resource Construction Set verwenden. Für diesen Fall wird ein Programm mitgeliefert, das jede Resource in einen C-Quelltext umsetzt.

Der Debugger von Laser C ist kein separates Programm, sondern wird als Objektdatei zu dem zu bearbeitenden Programm gelinkt. Er arbeitet symbolisch und ist ein gutes Hilfsmittel, um C-Programme zu entwanzen.

Ein Bonbon, das der Initialisierungscode der beiden Systeme enthält, ist die globale Variable "*_app*". Ist der Wert dieser Variablen ungleich Null, wurde das Programm als Applikation gestartet, sonst als Accessory geladen. Das Listing zeigt ein Beispiel, wie man sein Programm sowohl als Accessory als auch als Applikation verwenden kann. Man muß bei dem fertigen Programm lediglich die Extension auf "*.PRG*" bzw. "*.ACC*" setzen. Wie man sieht, sind die #include-Dateien nicht namensgleich. Je nachdem, welchen Compiler man verwendet, muß man die entsprechen Makros definieren, um die richtige Headerdatei einzubinden. Erstaunlich an dieser Erweiterung ist die Übereinstimmung des Namens der globalen Variablen. Wer hat da wohl von wem abgeguckt?

Fazit

Zunächst hört sich 198 DM für ein C-Entwicklungssystem wie Turbo C im Vergleich zu 398 DM für Laser C sehr viel günstiger an. Bedenkt man jedoch den Lieferumfang von Laser C, muß man zu Turbo noch das Assembler-/Debugger-Paket und ein Resource Construction Set kaufen. Wenn man an den Cache von Laser C denkt, kommt noch eine RAM-Disk dazu. Erst dann sind die beiden Pakete etwa auf dem gleichen Stand, auch, was den Preis angeht.

JL

Bezugsadressen:

Turbo C: Heimsoeth Software GmbH
Lindwurmstr. 88
8000 München 2

Laser C: Application Systems /// Heidelberg
Englerstr. 3
6900 Heidelberg

ENDE

AUF DER SCHWELLE ZUM LICHT



I/O-Umleitung

In den bisherigen Folgen haben wir die block- und zeichenorientierten Geräte stets getrennt besprochen. GEMDOS erlaubt aber, mit den *"Dateifunktionen"* *'Fopen'*, *'Fread'* usw. auch Devices wie Bildschirm und Tastatur anzusprechen. Ebenso lassen sich Dateien mit *'Cconin'*, *'Cconout'* usw. bearbeiten. Die realen Möglichkeiten hierzu sind jedoch relativ beschränkt, da zahlreiche Fehler einem das Leben schwer machen.

Heute werden wir intensiv auf einige bisherige Folgen zurückgreifen müssen. Daher könnte es sinnvoll sein, sich schon einmal die Ausgaben 6/88, 7/88 und 10/88 der ST-Computer herauszusuchen.

I/O-Umleitung - wozu?

In einigen Fällen möchte man die Ausgaben eines Programms ohne Programmänderungen auf verschiedenen Geräten ausgeben können. Zum Beispiel sollen Ergebnisse vielleicht nicht immer auf dem Bildschirm erscheinen, sondern auch wahlweise auf dem Drucker. Oder die Ausgabe soll sogar in eine Datei geschrieben werden, um sie später weiterzuverarbeiten.

Würden alle Ausgabefunktionen jeweils direkt bestimmte (physikalische) Geräte ansprechen, müßten für die Ausgabe auf Bildschirm und Drucker eigenständige Ausgaberoutinen programmiert werden.

Sinn der I/O-Umleitung (I/O steht für *"Input/Output"*, also *"Eingabe/Ausgabe"*) ist es also, die Ein- und Ausgabefunktionen des GEMDOS auf andere Geräte wirken zu lassen als die standardmäßig eingestellten.

So kann man *'Cconout'* dazu bringen, Zeichen in eine Datei zu schreiben. Analog kann man mit *'Fwrite'* auch auf dem Bildschirm oder Drucker ausgeben.

Ähnlich verhält es sich mit den Eingabefunktionen. So erlaubt z.B. *'Cconin'* auch die Eingabe von serieller Schnittstelle oder Datei, und *'Fread'* ermöglicht das Lesen von der Tastatur.

Ein weiterer Verwendungszweck ergibt sich bei Kommandointerpretern, die oft die Umlenkung der Ein- oder Ausgabe eines ganzen Programms (meist in eine Datei) erlauben. Hier stößt man jedoch relativ schnell an die Grenzen von GEMDOS.

Noch weitergehende Möglichkeiten, wie die *"Pipes"* des Betriebssystems UNIX, bei der die Ausgabe eines Programms direkt als Eingabe eines anderen verwendet wird, sind mit GEMDOS selbst nicht realisierbar.

Dateiende-Erkennung

Ein großes Manko der I/O-Umleitung des GEMDOS ist das Fehlen eines vernünftigen EOF-Konzepts. Bei Eingaben von einer Datei muß ein Dateiende (*EOF* = *"End of file"*) gesondert bemerkt werden, damit Programme wissen, wann sie keine weiteren Zeichen mehr erwarten können.

Einige Betriebssysteme benutzen hierzu ein spezielles Zeichen (*EOF-Marke*). Dieses Zeichen kann ein ganz bestimmtes *"normales"* Byte sein, das selbst in der Datei (als letztes Zeichen natürlich) steht.

Dabei muß gewährleistet sein, daß das EOF-Zeichen nicht in den Daten selbst vorkommen kann. Daher eignet sich dieses Verfahren hauptsächlich für Textdateien, nicht dagegen für Binärdateien (wie Programmdateien), in denen alle 256 verschiedenen Bytes durchaus vorkommen dürfen.

Eine andere Möglichkeit besteht darin, daß die Dateilänge vom Betriebssystem unabhängig verwaltet wird. Dann muß die Eingabefunktion dem Programm einen besonderen Wert zurückgeben, der eindeutig von legalen Eingabedaten unterschieden werden kann. Dies kann z.B. durch ein separates *EOF-Flag* geschehen. Wir gehen im weiteren von dieser Methode aus, da sie der des GEMDOS am ehesten entspricht.

Bisher war nur von Dateien die Rede. Da die Eingabefunktionen aber auch für zeichenorientierte Geräte wie die Tastatur verwendet werden können, müssen auch diese ein *"EOF"* kennen, damit Programme die Eingabedaten unabhängig vom Eingabegerät verarbeiten können.

Da mit diesen Geräten meist nur Text verarbeitet wird, kann hier ein besonderes Zeichen (z.B. *CTRL-Z = ASCII 26*) Verwendung finden. Das Betriebssystem gibt dann dieses Zeichen nicht an das Programm weiter (entfernt es also aus dem Eingabestrom) und meldet stattdessen den oben erwähnten besonderen *EOF-Status*, genau wie bei Dateien.

So weit, so gut. Nur, die Sache hat leider einen Haken. Es ist nämlich im GEMDOS so nicht realisiert. Genauer gesagt, hat GEMDOS eigentlich gar kein vernünftiges *EOF-Konzept*.

Bei den Dateifunktionen kann man mittels *'Fseek'* die Dateilänge ermitteln (s. 6/88), was schon ein wenig umständlich ist. Es ist auch möglich, solange (einzelne) Zeichen zu lesen, bis *'Fread'* 0L zurückgibt, um anzuzeigen, daß keine weiteren Zeichen mehr lesbar sind. Dies kann aber auch bei anderen (Lese-)Fehlern passieren, daher ist diese Methode nicht so zu empfehlen.

Bei Anwendung von *'Fseek'* auf ein zeichenorientiertes Gerät sieht es schon schlecht aus. *'Fseek'* liefert als Dateiposition und damit auch Dateilänge immer Null, und *'Fread'* wartet auf jeden Fall, bis ein Zeichen vorliegt. Es besteht keine Möglichkeit, z.B. über die Tastatur GEMDOS ein *"EOF"* zu signalisieren.

Bei den zeichenorientierten Funktionen sieht es noch schlimmer aus. Auf Devices angewandt, warten sie immer, bis ein Zeichen verfügbar ist (ausgenommen *'Crawio'*). Bei Dateien wird in jedem Fall ein Zeichen gelesen (auch bei *'Crawio'*). Beim Versuch, das Dateiende zu überschreiten, wird ein undefiniertes Zeichen zurückgegeben!

Zum Test, ob Zeichen vorliegen, gibt es die Funktionen *'Cconis'* und *'Cauxis'*. Bei Tastatur und serieller Schnittstelle zeigen sie an, ob schon Zeichen in den internen Puffern vorliegen oder ob auf eine Eingabe gewartet werden muß. Bei Dateien melden sie immer *"Zeichen verfügbar"*, so daß auch diese Funktionen zur *EOF-Erkennung* nutzlos sind.

Allein aus diesen Gründen verliert die Eingabeumleitung des ST einiges an Wert.

Bei der Umleitung der Ausgabe sieht es ein klein wenig besser aus, da das *EOF-Problem* entfällt. Aber auch hier macht GEMDOS einem das Leben schwer, wie schon in der letzten Folge am Beispiel des *"Hängens"* nach Umleitung der Ausgabe auf den Drucker deutlich wurde.

Und noch einmal: Handles

Fassen wir noch einmal unser Wissen über die *"Handles"* zusammen, das sich über mehrere Folgen verstreut angesammelt hat.

Device-Handles (-3...-1) sprechen direkt die zeichenorientierten Geräte (Bildschirm, Drucker, serielle Schnittstelle) an. Die Zuordnung zu den entsprechenden Devices auf BIOS-Ebene ist unveränderbar (10/88, Abb. 1 und 3).

Dateien bekommen beim Öffnen ein *Non-Standard-Handle* (6...80) zugewiesen, das außerdem einen *"File Control Block"* (FCB) bezeichnet, über den die internen Strukturen der Dateiverwaltung erreicht werden (6/88, Abb. 4 und 7/88, Abb. 5).

Mittels der I/O-Umleitung können auch Devices *Non-Standard-Handles* bekommen. Daher möchte ich mit Datei-Handles ganz speziell diejenigen Handles bezeichnen, die nach Berücksichtigung aller Umleitungsmöglichkeiten tatsächlich eine Datei bezeichnen.

Die GEMDOS-Funktionen *'C...'* operieren auf bestimmten Kanälen, die auch über *Standard-Handles* (0...5) erreichbar sind. Jedem Kanal ist entweder ein *Device-Handle* (der Normalfall) oder ein *Non-Standard-Handle* zugeordnet. Dieses Handle möchte ich mit dem Oberbegriff *Kanal-Handle* bezeichnen. Die übergeordnete Bezeichnung *Kanal-Handle* meint also das zu einem Kanal gehörende Handle und ist tatsächlich ein *Device-* oder *Non-Standard-Handle*. Das *Non-Standard-Handle* ist hier übrigens immer auch ein *Datei-Handle*.

Es sei daran erinnert, daß diese Kanal-Handles in der Tabelle *'p_devx'* im Prozeßdescriptor stehen (s. 10/88).

Da der FCB heute eine wichtige Rolle spielt, wird er hier nochmals erläutert (Abb. 1).

'f_fd' ist ein Zeiger auf den zugehörigen Filedescriptor (FD), wenn das *Non-Standard-Handle* ein *Datei-Handle* ist. Ein FCB kann jedoch auch ein Device bezeichnen. Dann ist *'f_fd'* das *Device-Handle* (das Byte ist auf 32 Bit vorzeichenerweitert, d.h. die Bits 31-8 sind alle gesetzt). NIL zeigt an, daß das Handle nicht belegt ist. Mit *'f_pd'* wird auf den Prozeßdescriptor des Prozesses verwiesen, der die Datei eröffnet hat. Auch hier steht NIL, wenn das Handle nicht belegt ist.

'f_cnt' zählt mit, an wieviele Aufrufer dieses Handle vergeben wurde. Ein Handle kann in Zusammenhang mit der I/O-Umleitung mehr als einmal vergeben werden. Beim Schließen von Dateien soll hiermit bemerkt werden, wann die Datei von allen Handle-Besitzern geschlossen wurde und FCB und FD somit freigegeben werden müssen.

Die FCBs sind statisch in einer globalen Tabelle *('fcbx')* organisiert, so daß mit einem Handle direkt auf den FCB zugegriffen werden kann. Das heißt, *'fcbx[handle-6]'* ist der FCB für *'handle'* (die Subtraktion von 6 erfolgt, da Non-Standard-Handles erst bei 6 anfangen).

Abb. 2 gibt zusammenfassend eine Übersicht der verschiedenen Handles. Dabei zeigen einfache Striche Gliederungen eines Begriffs an, und Pfeile weisen auf Abhängigkeiten hin, also wie z.B. aus einem Standard-Handle das zugehörige Kanal-Handle ermittelt wird.

Prinzipielle Möglichkeiten

Zunächst werden kurz die prinzipiellen Möglichkeiten der I/O-Umleitung besprochen, bevor es konkret um die GEMDOS-Funktionen selbst geht.

Direkt umgeleitet werden können nur die Kanäle. Einem Kanal kann ein anderes Device-Handle als das sonst übliche oder auch ein Non-Standard-Handle zugeordnet werden. Das Non-Standard-Handle bezeichnet dabei eine Datei.

Die sonst *"zeichenorientierten"* Funktionen *'C...'* werden somit auf ein anderes Device bzw. eine Datei umgelenkt.

Umgekehrt können bei den Dateioperationen 'F...' an Stelle der bisher betrachteten Non-Standard-Handles auch Standard-Handles übergeben werden, so daß über den Umweg der Kanäle auch die Devices erreicht werden können. Auch ist die direkte Angabe von Device-Handles möglich.

```
pedef struct
{   FD *f_fd;      /* Zeiger auf FD der Datei              */
    PD *f_pd;      /* Zeiger auf Prozeßdescriptor
                      des besitzenden Prozesses            */
    inf_cnt;       /* Anzahl der vergebenen Handles        */
}   FCB;
```

Abb. 1 - Der File Control-Block (FCB)

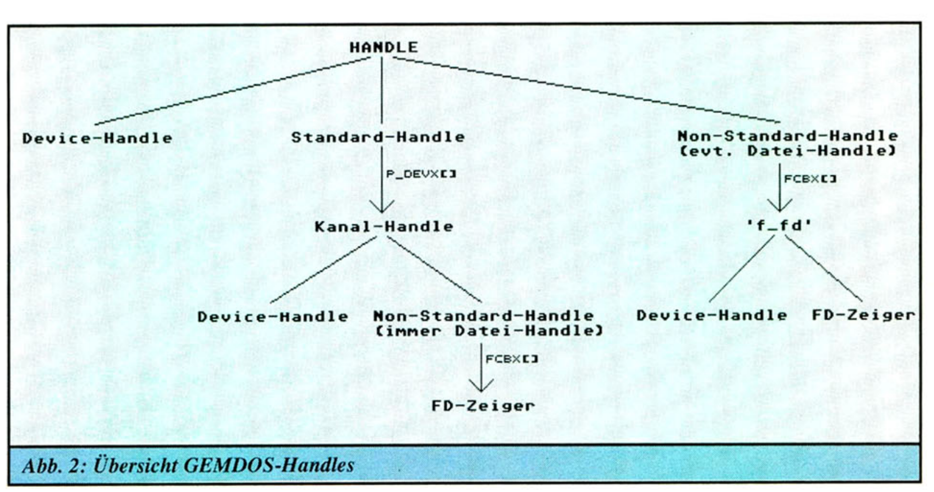


Abb. 2: Übersicht GEMDOS-Handles

Nun sind aber Situationen denkbar, in denen die Standardausgabe auf eine Datei umgelenkt wurde, aber trotzdem noch (über Dateifunktionen) angesprochen werden soll.

Daher besteht auch die Möglichkeit, für Kanäle ein Non-Standard-Handle generieren zu lassen. Das Device kann hiernach also sowohl mit dem üblichen Standard- als auch mit dem neuen Non-Standard-Handle angesprochen werden. Eine nachfolgende Umlenkung des Kanals betrifft nur das Standard-Handle, so daß das Device über das Non-Standard-Handle weiterhin erreichbar ist.

Außerdem kann bei der Umlenkung des Kanals statt des Device-auch das Non-Standard-Handle angegeben werden, was die Programmierung vereinheitlicht (Die Device-Handles müssen dem Programm nicht bekannt sein.).

GEMDOS-Funktionen zur I/O-Umleitung

Die eben erläuterten Möglichkeiten sind in den GEMDOS-Funktionen *'Fforce'* und *'Fdup'* realisiert.

Funktion $45 Fdup

int Fdup(int stdhnd)

Für das Standard-Handle *'stdhnd'* wird ein Non-Standard-Handle generiert, das Handle also quasi dupliziert. Das bisher über das Standard-Handle ansprechbare Device bzw. die Datei ist nun also zusätzlich auch noch über das neue Non-Standard-Handle ansprechbar.

Dabei wird der aktuelle Zustand des Kanals dupliziert. Dies bedeutet, daß spätere Umlenkungen des Kanals keinen Einfluß auf das die durch das Non-Standard-Handle angesprochene Device/Datei haben.

Rückgabewerte:

-35L (ENHNDL) kein Datei-Handle mehr frei
-37L (EIHNDL) 'stdhnd' ist kein Standard-Handle
6..80 Non-Standard-Handle, unter dem die Datei/das Device angesprochen werden kann.

Arbeitsweise

Nach der Überprüfung, ob *'stdhnd'* wirklich zwischen 0 und 5 liegt, wird in *'fcbx[]'* nach dem ersten freien Handle gesucht (erkannt an einem *'f_pd'*-Nullzeiger) und eventuell mit ENHNDL abgebrochen.

Der aktuelle Prozeß wird in *'f_pd'* eingetragen und *'f_cnt'* auf Eins gesetzt.

Wenn das durch *'stdhnd'* bezeichnete Kanal- (entnommen aus *'p_devx[]'*) ein Device-Handle ist (der "Normalfall"), wird dieses direkt als *'f_fd'*-"Zeiger" im FCB eingesetzt.

Ansonsten muß das Kanal- ein Non-Standard-Handle sein. Dann wird der zu diesem gehörende *'f_fd'*-Zeiger als *'f_fd'*-Zeiger für das neue Handle übernommen. Wenn das Non-Standard-Handle wieder ein Device bezeichnete, wird *'f_fd'* automatisch hier ebenfalls zum Device-Handle.

Falls das Non-Standard- aber ein Datei-Handle war, ist dessen FD nun über zwei Non-Standard-Handles ansprechbar, was zu diversen Konflikten führt (Diskussion siehe bei *'Fclose'* unten).

Funktion $46 Fforce

int Fforce(int stdhnd, int handle)

Der durch *'stdhnd'* bezeichnete Kanal wird auf die/das durch *'handle'* bezeichnete Datei/Device umgeleitet. Dabei kann *'handle'* ein Non-Standard- oder Device-Handle sein, wobei das Non-Standard-Handle selbst wiederum eine Device- oder Disk-Datei bezeichnen darf. *'handle'* darf nur kein Standard-Handle sein.

In allen mir bekannten GEMDOS-Dokumentationen ist nicht erwähnt, daß *'nsthnd'* auch Device-Handle sein kann. Dies vereinfacht die Umleitung eines Kanals, da kein *'Fdup'* notwendig ist, um ein Non-Standard-Handle für ein Device zu erhalten. Außerdem kann die Umleitung so mit einem einfachen *'Fforce'* wieder aufgehoben werden (an Stelle von *'Fclose'*, s.u.).

Unsinnige Kombinationen von Ein- und Ausgabeoperationen werden nicht erkannt. Die unzulässige Umlenkung des Standardeingabekanals auf Drucker wird also nicht bemerkt.

Rückgabewerte:

-37L (EIHNDL) *'stdhnd'* ist kein Standard-Handle oder *'handle'* ist Standard-Handle
0L alles ok.



STEUERN SPAREN MIT

STEUER TAX'88

MIT DEN NEUEN 88ER VORSCHRIFTEN

DAS UNENTBEHRLICHE PROGRAMM ZUR RICHTIGEN BERECHNUNG DER LOHN- UND EINKOMMENSSTEUER FÜR ALLE STEUERZAHLER MIT ST-COMPUTERN IN DER BRD UND WEST-BERLIN

VERSION 2.8

MIT DEN NEUEN STEUERLICHEN ÄNDERUNGEN UND VORSCHRIFTEN FÜR 1988

- voll unter GEM eingebunden
- mausgesteuert, einfache Bedienung
- auf allen ATARI-ST-Rechnern lauffähig (bei 260 TOS im ROM)
- Eingabe an Steuerformulare angepaßt
- Auswertung auf Monitor oder Drucker wurde dem Steuerbescheid angepaßt
- schnelles durcharbeiten, da durch Pull-down-Menues nur die erforderlichen Bereiche bearbeitet werden müssen
- mit vielen Hilfen, so daß auch der Laie mit seinem ATARI schnell und mühelos seine Steuer berechnen kann
- ausführliches Handbuch, somit systematische Einführung in das Steuerrecht, mit Steuertabellen und Tabellen für die Steuerklassenwahl bei Arbeitnehmer-Ehegatten
- ständig werden aktuelle Steuer-Tips aufgrund der Einkommensteuerrechtssprechung eingebaut
- dem Handbuch sind Musterformulare beigefügt, um z. B. Werbungskosten aus unselbständiger Tätigkeit geltend zu machen
- Update-Service für die Folgejahre
- alle Eingaben und Auswertungen können abgespeichert und später wieder aufgerufen werden, um zwischenzeitliche Änderungen einzugben und Neuberechnungen durchzuführen
- die Version 2.8 ist geeignet für den „normalen Anwender", der für sich seine Steuer berechnen will
- S/W oder Farbmonitor

DM 98,–

VERSION 3.8

MANDANTENFÄHIG

- Alle Merkmale wie Version 2.8, jedoch zusätzlich mit einer Datenbank. Programm deshalb mandantenfähig
- pro doppelseitiger Disk können ca. 250 Mandanten abgespeichert werden, auf 20 MB Harddisk ca. 6.600!
- die Version 3.8 eignet sich besonders – aber nicht nur – für Steuerberater, Lohnsteuervereine, Buchführungshelfer, Versicherungsvertreter usw., die die Steuer auch für andere berechnen oder aber für solche Anwender, die mehrere Fallbeispiele für sich durchrechnen und abspeichern wollen
- darüber hinaus auch für Selbständige sehr interessant, die mehrmals im Jahr bzw. ständig einen Überblick über ihre Steuerbelastung haben wollen, um z. B. Investitionsentscheidungen zu treffen, also nach dem Motto: was muß ich noch tun, um die Steuerbelastung zu drücken (was wäre wenn)

DM 159,–

UP-DATE SERVICE

STeuer Tax-Besitzer erhalten die neue Version 2.8 oder 3.8 gegen Rücksendung Ihrer registrierten Original-Diskette zum Preis von 35,– DM zuzügl. 5,– DM Versandkosten. Lieferung erfolgt nur gegen Übersendung eines Schecks in Höhe von 40,– DM.

DM 35,–

Alle Preise sind unverbindlich empfohlene Verkaufspreise.

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir: **Lohn- und Einkommensteuer-Programm**
______St. **STeuer-Tax '88** – Version **2.8** á 98,– DM
______St. **STeuer-Tax '88** – Version **3.8** á 159,– DM
______St. **STeuer-Tax '88** – UPDATE **2.8 / 3.8** á 40,– DM
zuzügl. Versandkosten 5,– DM unabhängig von der bestellten Stückzahl

Ich zahle:
☐ per Nachnahme
☐ per Scheck (liegt bei)

Name, Vorname ________________

Straße, Hausnr. ________________

PLZ, Ort ________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte

Heim Verlag

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Arbeitsweise

Wenn *'handle'* ein Device-Handle ist, wird dieses einfach zum neuen Kanal-Handle in *'p_devx[]'*.

Falls ein Non-Standard-Handle vorliegt, wird der FD-Zeger *'f_fd'* des FCB überprüft. Ein negativer Wert zeigt an, daß es sich um eine Device-Datei handelt. Also wird wieder nur das Device-zum neuen Kanal-Handle.

Im Falle einer Disk-Datei wird das Non-Standard-, das also ein Datei-Handle ist, selbst in *'p_devx[]'* eingetragen. Da nun der FD dieser Disk-Datei eine weitere Referenz bekommen hat (über das Standard-Handle), wird *'f_cnt'* inkrementiert.

Ein wesentlicher Fehler von *'Fforce'* ist, daß das bisherige Kanal-Handle unberücksichtigt bleibt. Denn wenn es schon ein Non-Standard-Handle war, müßte das zugehörige *'f_cnt'* wieder dekrementiert werden, um anzuzeigen, daß der FD nun einmal weniger referenziert wird. Im Extremfall, nämlich wenn *'f_cnt'* Null wird, müßte die Datei sogar automatisch geschlossen werden. Denn es wäre möglich, eine Datei zu eröffnen, dann einen Kanal darauf umzulenken, und dann *'Fclose'* aufzurufen. Da *'f_cnt'* noch auf 2 steht, wird die Datei nicht wirklich geschlossen (s. *'Fclose'*). Dies müßte aber nachgeholt werden, wenn die Kanalumlenkung rückgängig gemacht wird (dies geschieht spätestens bei der Programmterminierung).

Ein weiteres Problem ergibt sich beim Schließen von Dateien (s. unten bei *'Fclose'*).

GEMDOS-Datei Funktionen auf Devices & Kanälen

Bei der Besprechung der Datei-Funktionen 'F...' aus 6/88 bezogen sich alle Angaben stets auf Dateien, d.h. mit Handle war stets ein Datei-Handle gemeint.

Die meisten Funktionen erlauben auch die Angabe der anderen Handles bzw. der GEMDOS-Device-Namen *"CON:"*, *"AUX:"* und *"PRN:"*. Diese Fälle sollen hier betrachtet werden.

Funktion $3c Fcreate
Funktion $3d Fopen

*int Fcreate(char *path, int attr)*
*int Fopen(char *path, int mode)*

Das Device *'path'* wird *"eröffnet"*. Erlaubte Device-Namen sind *"CON:"*, *"AUX:"* und *"PRN:"*, wobei auch (durchgehende) Kleinschreibung möglich ist. *'attr'* und *'mode'* haben keine Bedeutung und sollten Null sein.

Das *"Öffnen"* dient nur der Ermittlung des Device-Handles, interne Strukturen werden nicht verändert. Dies ist etwas *"sauberer"*, als direkt die Device-Handles -1...-3 zu benutzen.

Im Gegensatz hierzu können Standard-Handles direkt verwendet werden. Daher brauchen und können Kanäle nicht geöffnet werden.

Rückgabewerte:

-3..-1 Device-Handle, unter dem die Datei angesprochen werden kann.

Arbeitsweise

Die beiden Funktionen machen nichts weiter, als die Device-Handles in Abhängigkeit vom Device-Namen zurückzugeben.

Funktion $3e Fclose

int Fclose(int handle)

Eine zuvor mit *'Fcreate'/'Fopen'* eröffnete Datei wird geschlossen. Dabei werden alle noch gepufferten Änderungen der Datei selbst und seines Directory-Eintrags auf das Medium geschrieben.

Mit *'Fopen'* *"geöffnete"* Devices können auch "geschlossen" werden, damit Devices genau wie Dateien behandelt werden können. Beim Schließen von Devices passiert allerdings effektiv gar nichts.

Auch Kanäle dürfen *"geschlossen"* werden. Dabei sollte eine eventuelle Umleitung rückgängig gemacht werden. In der aktuellen GEMDOS-Version ist der Kanal nach dem Schließen keinem Device mehr zugeordnet.

In einigen Fällen im Zusammenhang mit der I/O-Umleitung arbeitet *'Fclose'* fehlerhaft (s. *"Arbeitsweise"*).

Rückgabewerte:

-37L (EIHNDL) Handle ungültig
0L (EOK) alles ok.

Arbeitsweise

Bei Device-Handles wird sofort mit *0L* abgebrochen.

Das Schließen von Kanälen wird ziemlich unsinnig behandelt. Das Handle in *'p_devx[]'* wird nämlich einfach auf Null gesetzt und somit undefiniert! Wenn das Kanal-Handle ein Device bezeichnete, terminiert *'Fclose'* nun ebenfalls mit 0L.

Andernfalls liegt auf jeden Fall ein Non-Standard-Handle vor, entweder direkt als *'handle'* oder als Kanal-Handle aus *'p_devx[]'*. Wenn der zugehörige FD-Zeiger aus *'fcbx[]'* negativ ist, handelt es sich um ein Device, dem ein Non-Standard-Handle zugeordnet wurde. Demnach wird der *'f_cnt'*-Zähler des FCB dekrementiert. Erreicht er Null, wird der FCB nicht mehr gebraucht. Daher werden *'f_fd'* und *'f_pd'* gelöscht, so daß das Handle frei wird. *'Fclose'* beendet sich nun mit 0L.

Was jetzt noch bleibt, ist die Behandlung von Disk-Dateien. Dazu wird die Gültigkeit des Handles geprüft, eventuell wird hier mit EIHNDL abgebrochen.

Das eigentliche Schließen der Datei auf der Ebene der Dateiverwaltung übernimmt die interne Routine *'f_fclose'* (s. 6/88). Hier werden unter anderem gepufferte Sektoren zurückgeschrieben und der FD aus der internen FD-Liste ausgehängt (jedoch nicht an die interne Speicherverwaltung zurückgegeben).

Danach wird die Datei auf FCB-Ebene geschlossen. Dazu wird *'f_cnt'* um Eins erniedrigt. Wenn das Datei-Handle nur einmal vergeben war (*'f_cnt'* jetzt Null), wird der FD der internen Speicherverwaltung zurückgegeben, und *'f_fd'* und *'f_pd'* werden auf NIL gesetzt, so daß das Handle frei wird.

Die von *'f_fclose'* gelieferte Fehlermeldung wird nun zurückgegeben.

'Fclose' arbeitet in einigen Fällen fehlerhaft. Eine mehrfache Referenzierung des FD, wie es nach einem *'Fdup'* der Fall sein kann, sowie eine mehrfache Vergabe des Non-Standard-Handles, was *'Fforce'* unter Umständen macht, werden nicht richtig behandelt.

In beiden Fällen wird der FD auf jeden Fall noch gebraucht und darf daher weder aus der FD-Liste ausgehängt noch der Speicherverwaltung zurückgegeben werden.

'Fclose' sollte also in etwa so funktionieren:

Die FCB-Tabelle wird durchsucht, um Mehrfach-Referenzierungen des FD des zu schließenden Handles durch andere FCBs festzustellen.

Nun wird *'f_cnt'* dekrementiert. Wenn *'f_cnt'* Null erreicht hat und der FD nur

Dateifunkt. bei Devices		Zeichenfunktionen bei Dateien	
Fopen/Fcreate	+	Cconin,... (Zeicheneingabe)	(+)
Fclose	(+)	Cconout,... (Zeichenausgabe)	-
Fread	+	Cconws	(+)
Fwrite	+	Cconrs	+
Fseek	+	Cconis,Cauxis	-
Fdatime	-	Cconos,Cprnos,Cauxos	(+)

\+ *funktioniert offenbar*
(+) *funktioniert mit Einschränkungen*
\- *funktioniert überhaupt nicht*

Tab. 1: Funktionsfähigkeit der I/O-Umleitung

einmal referenziert wird, wird *'f_fclose'* so wie bisher aufgerufen, andernfalls derart, daß nur die Sektorpuffer zurückgeschrieben werden, aber der FD in der FD-Liste verbleibt. Im ersten Fall wird der FD anschließend der internen Speicherverwaltung zurückgegeben.

In jedem Fall wird jedoch, wenn *'f_cnt'* Null ist, der FCB freigegeben (durch Löschen von *'f_fd'* und *'f_pd'*).

Funktion $3f Fread

*long Fread(int handle, long count, char *buf)*

Es werden *'count'*-Zeichen aus der Datei *'handle'* in einen bei *'buf'* beginnenden Speicherbereich geladen. Dabei sind Device-, Standard- und Non-Standard-Handles erlaubt.

Bei Devices ist *'count'* auf 32 kB begrenzt.

Rückgabewerte:

-37L (EIHNDL) Handle ungültig
>=0 Zahl der tatsächlich gelesenen Zeichen, Vergleich mit 'count' gibt Auskunft, ob Fehler aufgetreten ist.

Arbeitsweise

Zuerst wird der zum Handle gehörige Zeiger auf den FD ermittelt. Bei Non-Standard-Handles steht er im FCB. Bei Standard-Handles wird das Kanal-Handle aus *'p_devx[]'* genommen. Wenn es ein Datei-Handle ist, wird der FD ebenfalls aus dem zugehörigen FCB geholt. Bei Kanal-Handles, die Devices bezeichnen, oder wenn 'handle' selbst schon Device-Handle ist, wird das Device-Handle selbst als "FD" genommen. Damit sind alle Fälle abgedeckt.

Wenn der mit dieser Prozedur ermittelte FD-Zeiger ein Null-Zeiger ist, wird mit EIHNDL abgebrochen.

Bei einem *"positiven"* Zeiger handelt es sich um eine Disk-Datei. Die Funktion von *'Fread'* bei Disk-Dateien wurde schon in 6/88 erläutert. Hier geht es um die Devices.

Bei einem *'count'*-Wert von 1 wird ein Zeichen wie mit *'Cconin'* vom entsprechenden Device gelesen. D.h. es wird solange gewartet, bis ein Zeichen vorhanden ist, und als Echo ausgegeben.

Größere 'count'-Werte veranlassen eine Eingabe ähnlich wie mit *'Cconrs'*. Allerdings beginnt *'buf'* hier nicht mit zwei besonderen Zeichen wie bei *'Cconrs'*.

Stattdessen wird *'count'* als Maximallänge genommen und die Anzahl der gelesenen Zeichen direkt von *'Fread'* zurückgegeben.

Bei *'count'*-Werten größer als 32 kB liest *'Fread'* nichts und gibt nur 0L zurück.

Funktion $40 Fwrite

*long Fwrite(int handle, long count, char *buf)*

Es werden *'count'*-Zeichen aus einem bei *'buf'* beginnenden Speicherbereich in die Datei *'handle'* geschrieben. Dabei sind Device-, Standard- und Non-Standard-Handles erlaubt. Bei Devices ist *'count'* auf 32 kB begrenzt.

Rückgabewerte:

-1..-31 BIOS-Fehlermeldung bei Diskzugriff
-37L (EIHNDL) Handle ungültig
>=0 Zahl der tatsächlich geschriebenen Zeichen, Vergleich mit *'count'* gibt Auskunft, ob Fehler aufgetreten ist.

Arbeitsweise

Genau wie bei 'Fread' wird der Zeiger auf den FD ermittelt und geprüft.

Wenn *"CON:"* das letztendlich resultierende Device ist, erfolgt die Ausgabe wie bei *'Cconout'*, d.h. CTRL-Codes werden erkannt und TAB (ASCII 9) zu Leerzeichen expandiert (8er-Tabulatoren).

Bei anderen Devices werden die Zeichen direkt mit BIOS-*'Bconout'* ausgegeben, es erfolgt also keine weitere Bearbeitung durch GEMDOS.

Da bei der Ausgabe i.allg. nichts schiefgehen kann, ist der Rückgabewert normalerweise immer gleich *'count'*.

Wenn *'count'* größer als 32 kB ist, wird nichts geschrieben. Es sollte somit 0L zurückgegeben werden. Bei einem *'count'*-Wert, der sogar größer als 64 kB ist, wird allerdings fälschlicherweise *'count'* selbst zurückgegeben.

Funktion $42 Fseek

int Fseek(long offset, int handle, int mode)

'Fseek' dient bei Dateien der Veränderung der aktuellen Dateiposition.

Handle darf Device-, Standard- oder Non-Standard-Handle sein. Bei Devices gibt es keine aktuelle Dateiposition. Daher passiert gar nichts, und es wird immer *0L* zurückgegeben.

Rückgabewerte:

-37L (EIHNDL) Handle ungültig
0L "neue Dateiposition"

Arbeitsweise

Die Ermittlung und Überprüfung des FD-Zeigers wird wie bei *'Fread'* durchgeführt. Bei Devices terminiert die Funktion sofort mit 0L.

Die anderen Dateifunktionen sind für Devices nicht zugelassen. Bei *'Fdelete'*, *'Fattrib'* und *'Frename'*, resultiert ein Device-Name als Pfadname in einer Fehlermeldung, vorausgesetzt es existiert keine Datei mit dem Device-Namen. Bei *'Fdatime'* ist die Reaktion mehr oder weniger undefiniert, da dort keine Überprüfung des Handles stattfindet (s. 6/88).

Ein einfaches Beispiel für die Anwendung der Dateifunktionen auf Devices zeigt Listing 1.

Zeichenorientierte GEMDOS-Funktionen für Dateien

Wie aus der letzten Folge bekannt ist, wirkt jede der *'C..'*-Funktionen auf eine feste Kanalnummer. In *'p_devx[]'* steht dann das Kanal-Handle, das das tatsächliche Ausgabegerät festlegt. Während wir letztes Mal nur den Fall von Device-Handles betrachtet haben, geht es heute um die Umlenkung eines Kanals auf eine

```
#include <osbind.h>

main()
{ int fh;
  char c;

  fh = Fopen("CON:",0);              /* Device "öffnen"        */
  Fwrite(fh,7L,"Hallo\r\n"); /* Text geht auf Konsole   */
  Fread(fh,1L,&c);           /* Eingabe auch von Konsole */
  Fclose(fh);                /* Device "schließen"      */
  Cnecin();                  /* auf Tastendruck warten  */
}
```

Listing 1: Dateifunktionen bei Devices

Disk-Datei. Das Kanal-Handle ist dann also stets ein Datei-Handle.

Funktion $01 Cconin
Funktion $03 Cauxin
Funktion $07 Crawcin
Funktion $08 Cnecin

long Cconin()
long Cauxin()
long Crawcin()
long Cnecin()

Ein Zeichen wird aus der Datei gelesen. Es wird vorzeichenerweitert als *'long'* zurückgegeben, d.h. bei Zeichen mit einem ASCII-Kode größer als 127 sind die Bits 31-8 gesetzt, ansonsten gelöscht.

Diese Funktionen liefern undefinierte Werte zurück, wenn ein Fehler beim Lesen auftritt oder das Dateiende überschritten wird.

Arbeitsweise

Das Lesen geschieht mit der internen Routine *'f_read'* (s. 6/88 und 3/88). Es wird aber nicht überprüft, ob *'f_read'* überhaupt erfolgreich war. Daher erklärt sich die Rückgabe undefinierter Werte. Ein vernünftiges Arbeiten ist also mit diesen Funktionen nicht möglich.

Funktion $02 Cconout
Funktion $04 Cauxout
Funktion $05 Cprnout

void Cconout(int c)
void Cauxout(int c)
void Cprnout(int c)

Ein Zeichen sollte in die Datei geschrieben werden. Diese Funktionen arbeiten überhaupt nicht und enden i.allg. mit zwei Bomben.

Arbeitsweise

Zum Schreiben wird die interne Routine *'f_write'* (s. 6/88 und 3/88) aufgerufen. Dabei wird jedoch die *"Pufferadresse"* falsch übergeben (fehlender *'C'*-Adreßoperator *'&'*), so daß es zum Absturz in *'f_write'* kommt.

Merkwürdigerweise wird außerdem noch der Rückgabewert von *'f_write'* (0 oder 1) zurückgegeben. Damit ließen sich zwar Schreibfehler erkennen, doch scheint dies wenig sinnvoll, da diese Funktionen bei Devices nichts zurückgeben, so daß eine einheitliche Benutzung im Sinne der I/O-Umleitung nicht möglich ist.

Funktion $06 Crawio

long Crawio(int c)

Wenn *'c'* den Wert $FF hat, wird *'c'* in die Datei geschrieben, ansonsten wird ein Zeichen gelesen. Dies geschieht genauso wie bei den oben erklärten Ein- und Ausgabefunktionen. *'Crawio'* unterscheidet sich bei Dateien also nicht von *'Cconin'* und *'Cconout'*.

Funktion $09 Cconws

*void Cconws(char *str)*

Der durch *'str'* adressierte und nullterminierte String wird in die Datei geschrieben. Das Nullbyte selbst wird nicht geschrieben und es erfolgt keine Ergänzung des Zeilenvorschubs (CR-LF).

Arbeitsweise

Alle Zeichen des Strings werden einzeln mit *'f_write'* ausgegeben. Daher ist *'Cconws'* relativ langsam (sofern man die geringe Geschwindigkeit der GEMDOS-Dateiverwaltung überhaupt noch für reduzierbar hält). Im Gegensatz zu den anderen Ausgabe-Funktionen ist die Parameterübergabe hier korrekt.

Schreibfehler werden allerdings nicht bemerkt. *'f_write'* wird in jedem Fall für jedes einzelne Zeichen aufgerufen.

Funktion $0A Cconrs

*void Cconrs(char *buf)*

Ein String wird aus der Datei gelesen. Der Aufbau von *'buf'* ist dabei genauso wie beim *'Cconrs'* von Device (10/88).

Die Eingabe bricht ab, wenn ein Lesefehler auftritt, das Dateiende erreicht wird oder ein CR (ASCII 13) gelesen wird. Im letzten Fall wird das nächste Zeichen (vermeintliches LF) auch noch gelesen (genauer: überlesen). Der String wird weder mit Null noch mit CR terminiert, einzig das in *'buf[0]'* abgelegte Byte gibt die Stringlänge an.

Außerdem erscheint die Eingabe auf der Standardausgabe (Kanal 1) als Echo (!), einschließlich des CR. Auch das Echo wird nicht mit LF abgeschlossen.

Die anderen Steuerzeichen des 'Cconrs' von Device werden hier nicht berücksichtigt.

Arbeitsweise

Zum Lesen wird *'f_read'* benutzt. Ein Rückgabewert von ungleich 1 führt zum Abbruch.

Sonst wird der GEMDOS-Dispatcher über eine spezielle *"Reentry-Prozedur"* erneut aufgerufen, um mit *'Fwrite'* auf Kanal 1 das Echo zu erzeugen. Anschließend erfolgt die Abfrage, ob CR gelesen wurde.

Die Funktion gibt immer 0 zurück, obwohl *'Cconrs'* eigentlich *'void'* ist (kein Kommentar!).

Funktion $0B Cconis
Funktion $10 Cconos
Funktion $11 Cprnos
Funktion $12 Cauxis
Funktion $13 Cauxos

long Cconis()
long Cconos()
long Cprnos()
long Cauxis()
long Cauxos()

Diese Funktionen geben immer $FF zurück, um *"Zeichen verfügbar"* bzw. *"Zeichen annehmbar"* zu signalisieren.

GEMDOS erkennt bei den Eingabestatus-Funktionen also nie das Dateiende, obwohl dies ohne großen Aufwand möglich wäre.

Außerdem ist die Rückgabe von $FF

sowieso nicht ganz korrekt, richtig wäre $FFFF (-1).

Listing 2 ist eine Demonstration der zeichenorientierten Funktionen bei Kanälen und zeigt noch einmal den Unterschied zwischen Device und Kanal.

So, das waren alle GEMDOS-Funktionen für Datei- und Device-Operationen noch einmal im Schnelldurchgang. Leider funktioniert hier vieles nicht so richtig. Tab. 1 gibt einen Überblick, welche Funktionen nach einer Umleitung noch getrost verwendet werden können und von welchen man lieber die Finger läßt.

Hinzu kommen noch die Fehler bei *'Fforce'* und bei der Umlenkung der Standardausgabe auf den Drucker (letzteres s. 10/88).

Ich muß allerdings gestehen, daß meine Programmiererfahrung mit der I/O-Umleitung sich in Grenzen hält, so daß es nicht unwahrscheinlich ist, daß es noch weitere Schwierigkeiten gibt.

Device oder Datei?

Wenn alle eben beschriebenen GEMDOS-Funktionen gleichermaßen bei Devices und Dateien funktionieren würden, könnte es dem Programm eigentlich egal sein, von wo z.B. *'Cconin'* seine Eingaben herholt.

Doch wegen der vielen Fehler und aus anderen Gründen (z.B. Effizienz) kann es sinnvoll sein zu wissen, ob sich hinter einem Handle letztendlich ein Device oder eine Datei verbirgt. Eine solche Funktion heißt in 'C' gewöhnlich *'isatty'*.

```
int isatty(handle)
int handle;
{ long pos, ret;

  pos = Fseek(0L,handle,1);     /* akt. Position merken      */
  ret = Fseek(1L,handle,0);     /* versuchen, Position 1 zu
                                   erreichen                 */
  Fseek(pos,handle,0);          /* zurück zu alter Position  */
  return (ret == 0L);         /* TRUE wenn 'handle' Device meint */
}
```

Listing 3: Beispiel-Implementation von 'isatty'

GEMDOS stellt dazu keine eigene Funktion bereit, doch mit einem kleinen Trick kommt man hier weiter. Die *'Fseek'*-Funktion läßt sich nämlich dazu mißbrauchen (Listing 3).

Das erste 'Fseek' dient nur dazu, die aktuelle Dateiposition zu merken. Das zweite *'Fseek'* versucht, die Dateiposition 1 anzuspringen. Bei Devices wird als *"neue Position"* immer 0L zurückgegeben, wie wir oben gesehen haben, bei Dateien im allgemeinen 1L. Eine Ausnahme bilden allerdings leere Dateien, bei denen die Fehlermeldung ERANGE zurückgegeben wird. 0L erhält man jedoch nur bei Devices, so daß eine eindeutige Unterscheidung möglich ist.

Anschließend wird die alte Dateiposition wieder ausgewählt (bei Devices passiert wieder nichts).

Auf diese Möglichkeit, *'isatty'* zu implementieren, ist übrigens auch ATARI gekommen, allerdings hat deren Beispiel-Listing zwei Fehler...

Man könnte auch auf die Idee kommen, *'isatty'* mit Hilfe von *'Fdatime'* zu implementieren, da diese Funktion ebenfalls nur für Dateien sinnvoll ist. Da *'Fdatime'* jedoch keine Überprüfung des Handles vornimmt und bei Device- oder Standard-Handles abstürzen kann, scheidet diese Möglichkeit aus.

```
#include <osbind.h>

main()
{ int fh;
  int ch;

  ch = Fopen("CON:",0);             /* Device-Handle holen        */
  fh = Fcreate("test",0);           /* Datei kreieren             */
  Fforce(1,fh);                     /* Standardausgabe -> Datei   */
  Cconws("Hallo1\r\n");             /* geht jetzt in Datei        */
  Fwrite(1,8L,"Hallo2\r\n");        /* auch dies geht in die
                                       Datei                      */
  Fwrite(ch,8L,"Hallo3\r\n");       /* dies aber auf den
                                       Bildschirm                 */
  Fforce(1,ch);                     /* Standardausgabe -> Konsole */
  Fclose(fh);                       /* Datei schließen            */
  Fclose(ch);                       /* Device "schließen"         */
  Cnecin();                         /* Tastendruck abwarten       */
}
```

Listing 2: Zeichenfunktionen bei Dateien

I/O-Umleitung & Tochterprozesse

Die Kanal-Handles sind lokal für einen Prozeß gültig, da die *'p_devx[]'*-Tabelle im Prozeßdescriptor (PD) liegt.

Daher erhebt sich die Frage, was mit der Belegung der Kanal-Handles geschieht, wenn der Prozeß gewechselt wird, sei es durch das Starten eines neuen Programms mit *'Pexec'* oder der Prozeßterminierung mit *'Pterm'*, *'Pterm0'* oder *'Ptermres'* (in Zukunft unter *'Pterm'* zusammengefaßt).

Genau wie die Standard-Pfade werden auch die Kanal-Zuordnungen an Tochterprozesse *"vererbt"*.

Ein einfaches Übertragen der Kanal-Handles in den Tochter-PD reicht allerdings nicht aus, da im Falle von Datei-Handles diese nun mehrfach vergeben sind. Deswegen wird beim *'Pexec'* für alle Kanal-Handles, die Datei-Handles sind, automatisch ein *'Fforce'* für den Tochter-PD durchgeführt. Dadurch wird gewährleistet, daß die *'f_cnt'*-Werte korrekt erhöht werden. Device-Handles werden direkt in den Tochter-PD kopiert.

Beim *'Pterm'* werden alle noch geöffneten Dateien des Prozesses geschlossen. Dabei sollte auch die automatische I/O-Umleitung durch das *'Pexec'* wieder rückgängig gemacht werden.

Dazu wird bei allen Kanal-Handles, die Datei-Handles sind, *'Fclose'* aufgerufen. Dies ist in Ordnung, wenn das Datei-

Handle vom terminierenden Prozeß selbst stammt, d.h. auf Grund eines vom Prozeß initiierten *'Fforce'* zum Kanal-Handle geworden ist. Falls es dagegen die Datei eines Parent-Prozesses bezeichnet, darf die Datei selbst nicht geschlossen werden. Dies ist aber auf Grund der fehlerhaften Arbeitsweise von *'Fclose'* leider der Fall. Dieser Effekt würde bei Beseitigung der Fehler von *'Fclose'* ebenfalls verschwinden (hoffe ich zumindest).

Anschließend wird noch die FCB-Tabelle nach FCBs durchsucht, die dem terminierenden Prozeß gehören. Deren zugehörige Dateien werden ebenfalls mit *'Fclose'* geschlossen.

Es ist wohl deutlich geworden, daß die Vererbung der Kanal-Zuordnungen ihre Tücken hat, so daß es wohl besser ist, vor einem *'Pexec'* dafür zu sorgen, daß alle Kanäle nur Devices ansprechen.

Außerdem gibt es noch einen weiteren *"Schönheitsfehler"*. Die automatische Umleitung des *'Pexec'* wird nämlich nicht beim Starten eines Programms durchgeführt, sondern schon beim Laden. Dies führt zu Schwierigkeiten, wenn ein Programm nur einmal geladen (mit *'Pexec'*-Modus 3 *"nur laden"*) und mehrmals gestartet werden soll (mit *'Pexec'*-Modus 4 *"nur starten"*).

Zwischen Laden und erstem Starten sollte der Parent-Prozeß daher keine Dateien schließen, die seinen Kanälen zugeordnet sind, da sie schon weitervererbt wurden.

Beim zweiten Start findet gar keine I/O-Umleitung mehr statt! Schlimmer noch ist, daß *'Pterm'* den *'p_devx[]'* unverändert läßt, so daß beim nächsten Start die Kanal-Handles Dateien bezeichnen, die schon längst geschlossen sind. Womöglich sind die Handles sogar schon wieder für neu eröffnete Dateien vergeben worden.

Daher muß man bei dieser Methode, Programme mehrfach zu starten, sicherstellen, daß der Tochterprozeß **keine** *'Fforce'*-Aufrufe macht und/oder das *'p_devx[]'*-Feld selbst mit geeigneten Standard-Werten (Device-Handles) versorgt wird.

Letzteres stellt aber einen unsauberen Eingriff ins Betriebssystem dar, da 'p_devx' nicht zu den von ATARI dokumentierten Teilen des PD gehört. Aber eine vollkommen legale Möglichkeit, Programme wirklich vernünftig resident zu laden, gibt es meines Wissens sowieso nicht.

Dieses Dilemma ist ja auch schon aus 7/88 bekannt, da sich die gleichen Probleme bei der Vererbung der Standard-Pfade ergeben.

Ausblick

Das traurige Kapitel der I/O-Umleitung ist nun beendet. Nächstes Mal werden wir uns den GEMDOS-Funktionen für Uhrzeit und Datum widmen.

Alex Esser

ENDE

CALAMUS

Der neue Stern am DTP-Himmel?

Nun ist er endlich da, lange angekündigt, viel gelobt, von vielen sehnsüchtig erwartet. Die vorliegende Version ist die kleine Ausführung von Calamus, die von ATARI selbst vertrieben wird.

Die Installation

Calamus befindet sich auf zwei Disketten und sollte, um vernünftiges Arbeiten zu ermöglichen, auf einer Harddisk installiert werden. Dazu muß man nur die Dateien und Ordner auf eine beliebige Partition kopieren, am besten in einen Ordner CALAMUS. Danach muß man in Calamus selber nur noch einige Zugriffspfade für Fonts, Grafiken usw. einstellen.

Das Konzept

Calamus ist ein rahmenorientiertes Layout-Programm. Das bedeutet, daß man auf einer Textseite Rahmen (rechteckige Kästen) verteilen kann, die für die Aufnahme der eigentlichen Layoutelemente wie Text, Grafik, Linien o.ä. bestimmt sind. In Calamus gibt es fünf Typen von Rahmen: für Text, Linien, Rasterflächen, Raster- und Vektorgrafiken. Textrahmen gibt es dabei noch in verschiedenen Varianten, für normalen Fließtext, Fußnoten oder auch gedrehten Text.

Rahmen dürfen sich beliebig überlappen. Selbstverständlich läßt sich die Reihenfolge, in der die Rahmen übereinanderliegen, verändern. Es ist aber auch kein Problem, einen Text mit einer Rasterfläche zu unterlegen. Um Bilder in einen Fließtext einfügen zu können, gibt es verschiedene Formatieroptionen, die den Text um einen darüberliegenden Rahmen 'herumfließen' lassen. Schließlich können Rahmen beliebig zu Gruppen zusammengefaßt und dann gemeinsam bearbeitet werden.

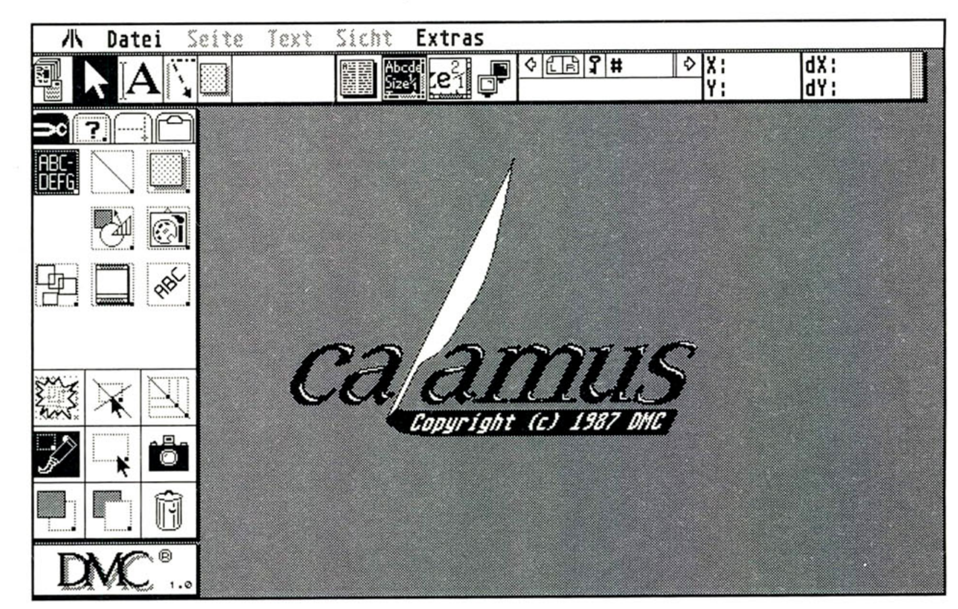
Die Gesamtheit aller leeren Rahmen einer Seite wird in Calamus als *'Seitenlayout'* bezeichnet. Seitenlayouts lassen sich beliebig kopieren, so daß es ein leichtes ist, aus einer *'Prototyp-Seite'* mehrere gleichartige Seiten zu generieren. Da es aber auch immer Elemente gibt, die auf jeder Seite den gleichen Inhalt haben sollen (zum Besipiel Kopf- oder Fußzeilen), lassen sich beliebig viele Rahmen von einem beliebigen Typ als Stammelemente definieren. Auch Seiten- und Kapitelnummern können in Stammrahmen eingefügt werden.

Das Maß aller Dinge ist in Calamus selbstverständlich einstellbar: Zentimeter, Pica, Cicero oder Inches stehen zur Auswahl, Fonts können in *Punkt pica*, *Punkt cicero* oder *mm* gemessen werden.

Die Position der Maus auf einer Seite wird immer auf dem Bildschirm angezeigt, natürlich in der gewählten Maßeinheit.

Die Bedienung

Calamus wird vorwiegend über eine grafische Benutzerführung mit vielen, vielen Icons in einer recht umfangreichen Hierarchie gesteuert. Die Menübox liegt wahlweise am rechten oder linken Bildrand, die allerwichtigsten Funktionen sind jedoch über eine Leiste unter der Menüleiste zu erreichen. Man kann die Menübox auch unter einem Textfenster verschwinden lassen und Calamus über eine Unzahl von Tastaturkommandos steuern. Ein Hoch dem guten Gedächtnis. Die Aufteilung der Programmfunktionen in Icons und Hierarchieebenen ist größtenteils gut und übersichtlich, manchmal ist es aber doch etwas zuviel, und man wünscht sich das eine oder andere Bildchen weniger. In einem Menü hat ein Icon zum Beispiel je nach Modus verschiedene Funktionen, was einen unerfahrenen Benutzer schon einmal in schwere Verwirrung stürzen kann. Glücklicherweise gibt es eine kleine Hilfefunktion, die in



der rechten oberen Bildschirmecke im Klartext die Bedeutung jedes einzelnen Icons anzeigt. Sonst wäre man wohl oft hoffnungslos verloren.

Die Calamus Module

Für den Benutzer ist Calamus in fünf Hauptmodule aufgeteilt: Seitenmontage, Rahmenfunktionen, Textgestaltung sowie Linien- und Rasterflächenmodul.

Die Binderei Seitenmontage

Das Seitenmontagemodul dient, wie der Name schon sagt, dazu, einzelne Seiten zu erzeugen sowie fertige Seiten (Rahmen mit Inhalt) oder Layouts (Rahmen ohne Inhalt) zu kopieren, zu löschen oder zu verschieben. Außerdem können sowohl einzelne Seiten als auch einzelne Layouts auf Diskette gespeichert oder von dort geladen werden.

Immer der richtige Rahmen

Das Rahmenmodul ist für die Erzeugung und Bearbeitung der Rahmen zuständig. Hier steht eine Unzahl von Funktionen sowohl allgemeiner Art (für alle Rahmentypen) als auch speziell für bestimmte Rahmentypen zur Verfügung: Gruppen können gebildet und wieder aufgelöst werden, die Definition von Stammelementen wird ebenfalls hier durchgeführt. Rahmen können natürlich kopiert (mit Inhalt - hierfür gibt es sogar ein Klemmbrett mit fünf ‘Plätzen!) und gelöscht werden, es ist aber auch möglich, einen Rahmen vor versehentlichen Veränderungen zu schützen.

Natürlich gibt es Hilfen fürs Rahmenbasteln: Während man einen Rahmen öffnet, wird, zusätzlich zu der absoluten Mausposition auf der Seite, auch die Größe des Rahmens in beiden Richtungen angezeigt. Außerdem gibt’s ein Lineal, man kann Hilfslinien und Hilfsraster, auch speziell für Spaltensatz, setzen, beides auch noch ‘magnetisieren’ (also Rahmen automatisch auf eine Hilfslinie ‘einrasten’ lassen), der Mauscursor kann sich in ein Fadenkreuz verwandeln - kurz, hier gibt es alles, was das Herz begehrt.

Natürlich kann man auch einzelne Rahmentypen fürs Edieren an- und ausschalten, um ein schnelleres Arbeiten zu ermöglichen. Um einen guten Eindruck von einer fertigen Seite zu bekommen, kann man schließlich auch die Umrandung eines Rahmens ausschalten und die ganze Schönheit des Layouts sozusagen unbedeckt genießen.

Spezialfunktionen für Textrahmen

Die wichtigsten Spezialfunktionen im Rahmenmodul dienen der Beeinflussung von Textrahmen. Hier kann bestimmt werden, ob ein Rahmen normalen Text enthält und frei ediert werden kann, oder ob es sich um einen Fußnoten- oder Indexrahmen handelt. Die Bedeutung von Fußnotenrahmen ist klar: Hier werden Fußnoten plaziert. Leider geht das nicht ganz so vor sich, wie man es von einer richtigen Fußnotenverwaltung erwarten würde: Man muß einen Rahmen von fester Größe für die Fußnoten vorgeben, Calamus kann die Größe dieses Rahmens nicht nach Bedarf anpassen. Einen Textausschnitt zur Fußnote zu machen, ist dagegen einfach: Einfach den gewünschten Text in einem Textrahmen markieren und per Mausklick auf ein Icon in die Fußnote schicken. Eine Fußnotennummer wird dabei automatisch in den Text eingefügt; die Numerierung ist recht flexibel. Dennoch, für wissenschaftliche Dokumente, die traditionell viele Fußnoten besitzen, ist diese Art der Fußnotenverwaltung nicht geeignet. DMC arbeitet, wie man mir auf Anfrage mitteilte, jedoch bereits an dem Problem.

Indexrahmen sind etwas ganz ähnliches, jedoch sind sie, wie der Name schon vermuten läßt, für Indizes gedacht. Auch hier

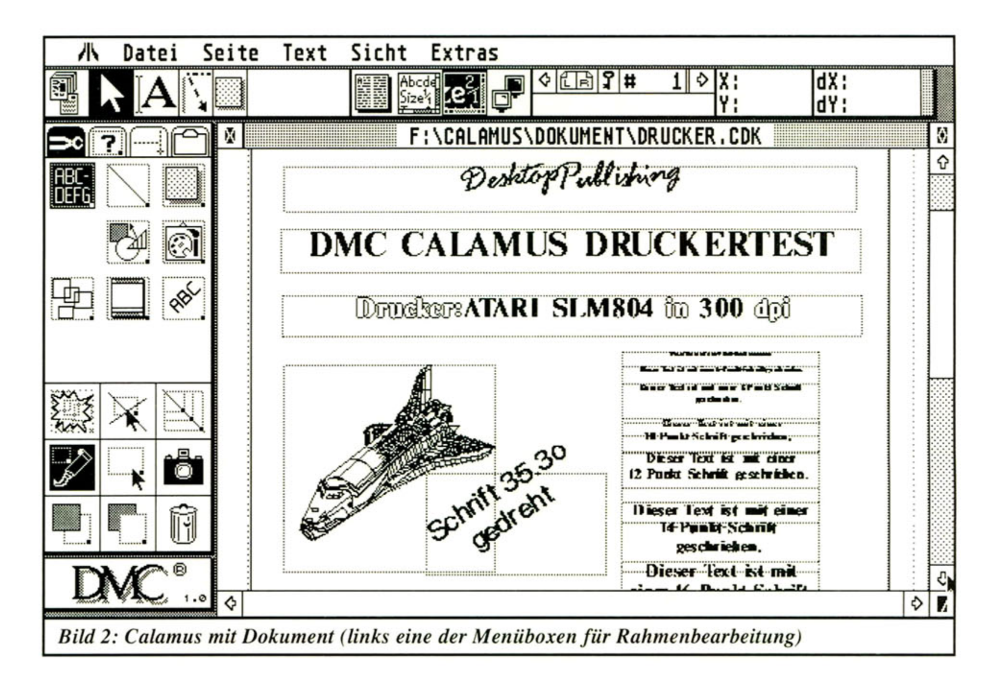


Bild 2: Calamus mit Dokument (links eine der Menüboxen für Rahmenbearbeitung)

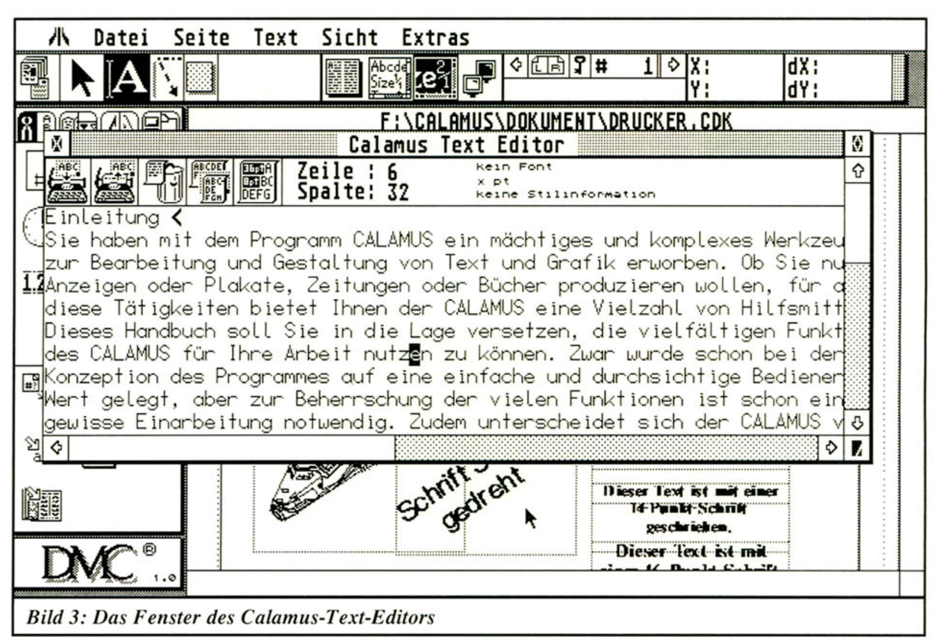


Bild 3: Das Fenster des Calamus-Text-Editors

wählt man per Maus einen Text aus und schickt ihn in einen Indexrahmen, jedoch muß dieser sich nicht auf der gleichen Seite befinden, sondern kann, wie es sich für einen Index gehört, zum Beispiel auf einer der letzten Seiten sein. Im Gegensatz zur Fußnotenverwaltung wird hier der markierte Text nur wahlweise aus dem Originalrahmen gelöscht. Im Augenblick ist die Indexverwaltung insofern nicht sehr nützlich, als sie keine selbständige alphabetische Ordnung erlaubt. Im Handbuch steht, daß spätere Versionen diesen Mangel korrigieren werden. Hoffen wir, daß es bis dahin nicht so lange dauert wie bis zur ersten Version.

Die übrigen Spezialfunktionen für Textrahmen beschäftigen sich mit dem Textfluß. Damit ist zum einen die Möglichkeit gemeint, Text um über dem Textrahmen liegende Bilder herumfließen zu lassen. Dabei kann man wählen, ob der Text beidseitig oder nur rechts bzw. links am oberen Rahmen vorbeifließen soll. Das Problem, Bilder in einen mehrspaltigen Text einzufügen, deren Format nicht der Spaltenbreite entspricht, dürfte damit der Vergangenheit angehören.

Die andere Art der Textflußfunktionen beschäftigt sich mit der Verbindung mehrerer Rahmen zu einer Textflußkette. Wenn man eine mehrspaltige Seite setzen will, ist es sehr mühselig, einen Text genau auf Spalten aufzuteilen. Also verbindet man alle Spalten einer Seite zu einer Kette, und der Text fließt automatisch in den nächsten Rahmen, wenn er nicht mehr in den gerade aktuellen hineinpaßt. Löscht man nun in einem Rahmen, der in der Kette weiter vorne liegt, fließt der Text zurück auf den freigewordenen Platz. Calamus bietet eine ganze Reihe von Funktionen, mit denen Textflußketten gebildet, angezeigt oder gelöscht werden können. Auch einmaliger Textabfluß von einem Rahmen in einen anderen ist möglich, wenn Rahmen einmal nicht zu einer Kette verbunden werden sollen.

Texte lassen sich um einen beliebigen Winkel verdrehen; allerdings muß man sie dazu in einen besonderen Rahmentyp für gedrehten Text bringen. Mit den Spezialfunktionen für diesen Rahmentyp läßt sich dann ein Drehwinkel einstellen und verändern.

Spezialfunktionen für Raster- & Vektorgrafikrahmen

In der jetzigen Ausführung von Calamus kann man keine Vektor- oder Rastergrafiken erzeugen; dies bleibt der Anfang '89 erscheinenden Profi-Version vorbehalten. Es ist jedoch möglich, fremde Grafiken einzulesen. Dabei kann Calamus zahlreiche File-Formate lesen: Bei Vektorgrafiken sind ein DMC-eigenes und das GEM-Metafile-Format implementiert, so daß auch auf dem PC erstellte Vektorgrafiken übernommen werden können. Rastergrafiken können ebenfalls in DMC-eigenem Format oder aber in Degas-, STAD-, STAD-Block-, GEM-Image-, GFA-Block- oder IFF-/Degas-Blockformat gelesen werden.

Die Spezialfunktionen für Vektorgrafik sind sparsam: Rahmen dieses Typs können lediglich auf eine 'Idealgröße' gebracht werden, die den Originalproportionen der Vektorgrafik entspricht.

Vielseitiger ist man da mit Rasterbildern: Man kann immerhin einen Namen für ein Bild vergeben, einen Bildteil ausschneiden und die Rahmengröße so optimieren, daß sich für Bildschirm oder Drucker (abhängig vom gewählten Treiber) eine optimale Darstellungsqualität ergibt.

Das Text-Modul

ist vielleicht das wichtigste in einem Layout-Programm. Calamus verfügt über eine Vielzahl von Möglichkeiten für die Textdarstellung. Die wichtigste Frage bei einem DTP-Programm ist aber zunächst, wie man den Text überhaupt in das DTP-Programm hineinbekommt. Zum einen kann man Textrahmen, genau wie die Grafikrahmen, mit einer Import-Funktion durch 'fremde' Texte füllen lassen. Calamus versteht das 1st_Word plus-Format sowie dankenswerterweise verschiedene ASCII-Formate, die sich durch die Interpretation der Linefeed- und Return-Zeichen am Zeilenende unterscheiden.

Durch die Wahl des richtigen Formates kann man sich so im allgemeinen unerwünschte Absätze nach jeder Zeile, die man sonst mühselig von Hand entfernen müßte, vom Leibe halten. Erwähnenswert ist noch, daß Calamus beim Einlesen von 1st_Word-Texten jedem 1st_Word-Schriftattribut einen eigenen Schrifttyp zuordnen kann. Ärgerlich ist dabei allerdings, daß Calamus aus jeder 1st_Word-Zeile einen eigenen Absatz macht.

Außerdem hat man aus den Schwächen diverser Konkurrenzprodukte gelernt und einen eigenen schnellen Texteditor integriert, der es auch möglich macht, Text direkt in Calamus zu erfassen. Ein guter Teil der Calamus-Funktionen steht auch im Editor, der sich übrigens komplett in einem Fenster befindet, zur Verfügung. Das Fensterprinzip hat beim Texteditor den großen Vorteil, daß man nicht unbedingt zwischen Layout und Editor umschalten muß, sondern, in gewissen Grenzen, mit beiden parallel arbeiten kann. So richtig zum Tragen kommt dieser Vorteil allerdings nur auf einem 19"-Großbildschirm, wie der Verfasser voller Begeisterung feststellen konnte, nachdem ein solches Wundergerät zwecks Test auf seinen Schreibtisch verschlagen wurde. An dieser Stelle daher gleich der Hinweis: Wer sich ernsthaft oder halbernsthaft mit DTP beschäftigen will, für den ist ein Ganzsei-

Datei Seite Text Sicht Extras Im Wörterbuch blättern
F:\CALAMUS\DOKUMENT\DRUCKER.CDK
bür-ste
bür-sten
bü-schel
ca-dil-lac
ca-fe
ca-mem-bert
camp
cam-pen
cam-ping
campt
camp-te
camp-ten
Ca-la-mus
OK
0.0 pt
0.0 pt
DMC

Bild 4: Das Fenster des Calamus-Wörterbuchs

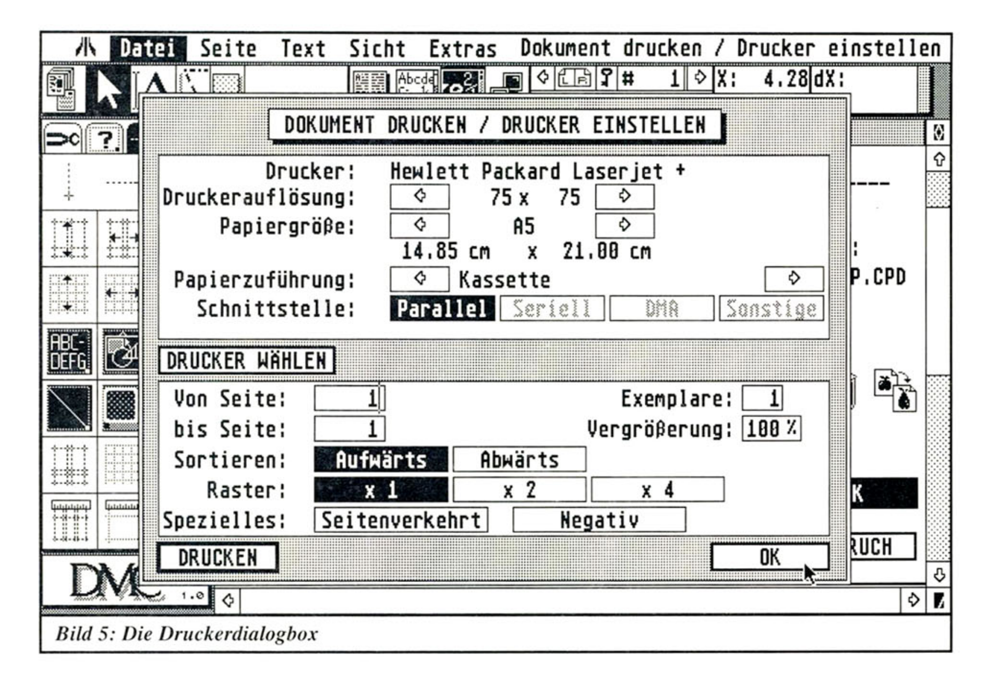


Bild 5: Die Druckerdialogbox

tenmonitor ein absolutes Muß. Aber zurück zum Editor, das Modul ist schnell und gibt den Text gut lesbar wieder, da glücklicherweise nicht der ATARI-Zeichensatz verwendet wird. Steuerzeichen für Schriftart, Schriftgröße usw. werden als fettgeschriebene Steuerzeichen (wahlweise im Klartext!) per Mausklick in den Text eingefügt, der Text wird immer im Flattersatz dargestellt. Wie häufig man den Editor benutzen wird, hängt, wie ich glaube, stark von der persönlichen Arbeitsweise ab; ich persönlich würde längere Texte lieber mit Tempus eingeben. Aber das ist, wie gesagt, nur Geschmacksache.

Übrigens existieren auch 'Suchen & Ersetzen' (für Text- und Schriftart bzw. Stil) sowie Blockfunktionen, beide arbeiten sowohl im Editor als auch auf der Layout-Seite, der Editor bietet also wirklich alles, was ein Editor braucht.

Schriften

Nach der Texteingabe ist das nächste wichtige Problem die Schrift, genauer gesagt die Typographie. Calamus bietet in dieser Richtung einiges, nämlich Vektorfonts. Vektorfonts bestehen nicht, wie Sie es von den üblichen Drucker- oder Bildschirmfonts kennen, aus einzelnen Punkten, sondern sie werden durch Linienzüge beschrieben. Theoretisch führt dies zu einer weitaus höheren Qualität, in der Praxis gibt es da aber ein paar Fallstricke. Da Vektorfonts aus Linien bestehen, kann man sie leicht vergrößern, ohne daß das Problem mit der groben Auflösung entsteht, das man ja von allen Rasterfonts kennt. Daher ist die große SIGNUM-Schrift ja auch schlechter als die kleine. Leider sind Vektorfonts in kleinen Schriftgrößen bei verhältnismäßig niedrig auflösenden Ausgabegeräten aber gröber als Rasterfonts. Der Grund dafür ist, daß man das Erscheinungsbild eines Rasterfonts per Hand auf die Auflösung des Ausgabegerätes optimiert, indem man Punkte hinzufügt, wo eigentlich keine hingehören, und umgekehrt. Offensichtlich geht dies bei Vektorfonts nicht, denn deren Linien werden ja immer mit einem festen Algorithmus in Rasterpunkte umgesetzt. Sie können also die Auflösung eines Druckers im Grenzberich nicht so gut ausnutzen wie Rasterfonts. Mit diesem Problem hat auch Calamus zu kämpfen - auf Matrix- und Laserdruckern ist die Qualität bei kleinen Schriftgrößen (6 und 8 Punkte) schlechter als z.B. bei SIGNUM. Erst auf einem Laser-Belichter mit seiner erheblich höheren Auflösung können Vektorfonts ihre Überlegenheit

kann, sowie die Textformatierung. Rechts- und Linksbündig, Blocksatz und Zentrierter Satz sind vorgesehen. Der Blocksatz funktioniert manchmal nicht auf Anhieb, im allgemeinen klappt es aber beim zweiten Versuch.

Schließlich kann man auch noch den Absatz-Abstand einstellen. Bei der Formatierung wird ein automatisches Unterschneiden vorgenommen, dessen Wirkung jedoch auch für einen Textlineal-Bereich verändert werden kann (Sperrung!). Selbstverständlich existiert jedoch auch die Möglichkeit, das Unterschneiden in jeder Richtung (links/rechts/oben/unten) manuell zu korrigieren. So wird zum Beispiel Formelsatz möglich. Leider ist das manuelle Unterschneiden im Augenblick recht bedienerunfreundlich, da nach jeder Veränderung der komplette Bildschirm neu aufgebaut werden muß, was bei einer Textseite doch etwas dauert. Da lobe ich mir die Lösung, die der Publishing-Partner bietet: Das Unterschneiden wird dort in einer Dialogbox eingestellt, in der man nur die zwei beteiligten Buchstaben sieht. Das geht schneller und ist erheblich komfortabler.

Auch der mittlere Wortabstand ist zwischen zwei Textlinealen variierbar. Skurrilerweise sind die beiden letzten Funktionen (Variation des Buchstaben- bzw. Wortabstandes) in einer anderen Menübox gelandet, was nicht der Übersicht zugute kommt.

Trennung

Ein sehr kritischer Punkt bei der Betrachtung einer Textformatierung ist die automatische Trennung. Calamus trennt nach Regeln, benötigt allerdings für vernünfti-

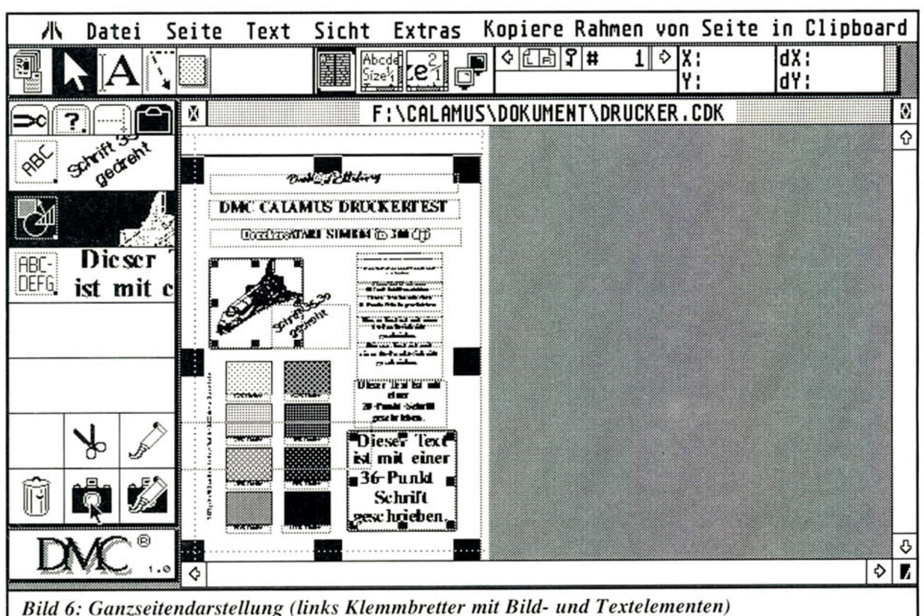


Bild 6: Ganzseitendarstellung (links Klemmbretter mit Bild- und Textelementen)

ge Ergebnisse ein Ausnahmelexikon, das zwar mitgeliefert, aber nicht ausreichend ist; man kann es jedoch leicht selbst erweitern. Die ganze Trennfunktion arbeitet brauchbar, wenn auch langsam, ist aber nicht überragend bedienerfreundlich und im Handbuch nicht vollständig beschrieben. Außerdem soll bis zum Ende des Jahres eine völlig neue und auch modulare (für verschiedene Sprachen) Trennfunktion zur Verfügung stehen. Wörterbücher kann man natürlich speichern und laden.

Sonstiges für Text

Bereits oben wurde die Möglichkeit beschrieben, Textteile in Fußnoten- oder Indexrahmen umzuplazieren. Wie gesagt bedarf beides noch der Verbesserung, um wirklich nützlich zu sein.

Es gibt noch einige Einfügeoptionen, nämlich für aktuelle Seitennummer, folgende Seitennummer, Kapitelnummer (mit vielen Optionen für die Zählweise: römisch oder arabisch, mit Buchstaben usw...), aktuelles Datum und aktuelle Uhrzeit. In einer Zeichensatzübersicht kann ein beliebiges Zeichen ausgewählt und in den Text übernommen werden.

Für Text gibt es fünf Klemmbretter, die das Kopieren und Verschieben von Textausschnitten erheblich verbessern. Wie alle Klemmbretter von Calamus erleichtern auch diese das Leben sehr, da man immer die linke obere Ecke des im Klemmbrett befindlichen Textes sehen kann.

Schließlich gibt es für die Textformatierung programmier- und speicherbare Makros, mit denen man das Einstellen häufig benutzter Textformate, Schriftarten, Lineale usw. stark beschleunigen und optimieren kann.

Seitenformate

Nachdem ich jetzt in Formatiermöglichkeiten für einen Textrahmen schwelgte, wird es Zeit, sich auch einmal mit den Seitenformaten zu beschäftigen. Hier bietet Calamus vordefinierte Formate, nämlich A5, A4, A3 und B5 sowie die amerikanischen *'Letter'*-, *'Legal'*-, *'Double'*- und *'Half'*-Formate. Außerdem kann man ein eigenes Format bestimmen sowie zwischen Hoch- oder Querformat wählen.

Zusätzlich können Sie auswählen, ob Ihr Dokument ein- oder doppelseitig sein soll, selbstverständlich mit unabhängigen Rändern. Dabei können, sehr praktisch, eventuelle Randbeschränkungen aus der Druckertreiberdefinition übernommen werden.

Das Linien- & das Rasterflächenmodul

Ein Calamus-Linienrahmen enthält immer nur eine Linie. Dabei stehen 18 verschiedene Formen zur Auswahl: gerade, gebogene, diagonale und geknickte Linien. Sie können die Linien mit einem Schatten von wählbarer Position und Füllung versehen sowie natürlich auch Linienstärke, Linienart (gestrichelt, durchgezogen usw.), Linienende (Pfeil, rund, kantig) und Linienfarbe (schwarz oder weiß) bzw. Füllmuster festlegen.

Variable Bildschirmdarstellung

Calamus kann eine Textseite so verkleinern, daß sie komplett auf den Bildschirm paßt. Man kann dann zwar nicht mehr viel lesen, aber bekommt einen Eindruck vom Gesamtlayout. In Normaldarstellung wird ein Seitenausschnitt in Druckergröße gezeigt. Auch doppelte 'Originalgröße' ist möglich, eine weitere Option erlaubt es, einen beliebigen anderen voreingestellten Wert zu verwenden. Der Bildaufbau ist ähnlich schnell/langsam wie bei der Konkurrenz, sehr groß ist der Unterschied nicht. Manchmal muß man eben warten.

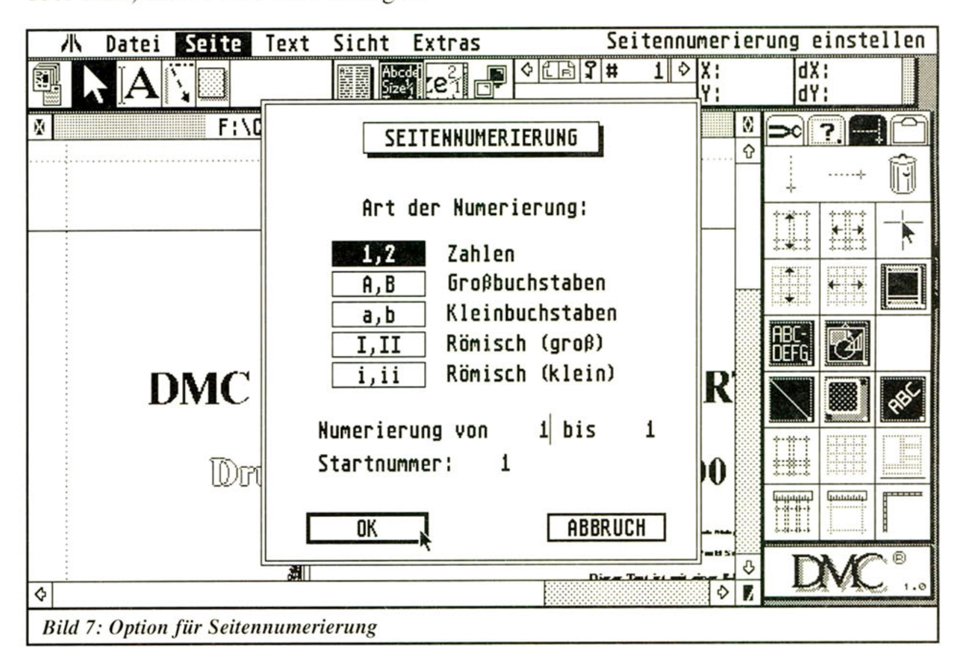


Bild 7: Option für Seitennumerierung

Das Rasterflächenmodul funktioniert genau wie das Linienmodul. Sie wählen eine von 15 Grundformen für das Raster (Viereck, Kreis, Stern, Raute, Dreieck usw.) und können sie mit Schatten und Füllmuster sowie einer Umrandung versehen. Einfach aber wirksam. Rasterbilder können übrigens auch mit einem Scanner eingelesen werden; drei weitverbreitete Scanner werden im Moment unterstützt (Silver Reed/Hawk/Panasonic).

Die Flexibilität der beiden Module läßt im Vergleich mit den sonstigen Möglichkeiten des Programmes etwas zu wünschen übrig. So ganz wird man das Gefühl, es handelt sich um eine Notlösung, nicht los, weil der Vektor- bzw Rastergrafikeditor, der dereinst die Profiversion schmücken soll, hier fehlt. Allerdings sollen die Module auch in der Profiversion erhalten bleiben.

Praxis & Sicherheit

Calamus ist inzwischen bei Version 1.01 angelangt, die sich von der 1.0-Version durch beseitigte Fehler unterscheidet. Dies ist höchst erfreulich, begeisterte doch die 1.0-Version durch ihre Fähigkeit zu spektakulären Abstürzen (Sie trieb mich beinahe zur Verzweiflung...). Nun, die 1.01-Fassung ist erheblich sicherer. Sie stürzt zwar immer noch ab, aber bei weitem nicht mehr so oft. Die Trennhilfe (bzw. das Wörterbuch) ist allerdings immer noch anfällig. Mit der 1.01-Version kann man wohl arbeiten.

Im praktischen Einsatz zeigt sich, daß Calamus recht gut konzipiert ist. Die meisten Features sind leicht erreichbar, etwas störend ist nur, daß man sich immer durch sehr viele Menüs hangeln muß. Aber das ist eigentlich kein Problem von Calamus, sondern von ATARI, so ein

12"-Bildschirm ist einfach zu klein. Schade ist nur, daß DMC in der Version für den Großbildschirm nicht mehr Platz für die Menüs spendiert hat, aber dann müßte es eben eine völlig andere Programmversion für den Großbildschirm geben. Vielleicht kommt so etwas ja eines Tages, wenn man bei DMC Zeit dafür findet.

Im Vergleich mit der Konkurrenz muß man feststellen, daß Calamus nicht in allen Punkten so eine überlegene Qualität darstellt, wie es nach den Vorschußlorbeeren den Anschein hatte. Der Hauptunterschied im Konzept, die Verwendung von Vektor- statt Rasterfonts, bringt im nichtprofessionellen Bereich (d.h. bis zum Laserdrucker) bei den meistbenutzten Schriftgrößen zwischen 8 und 12 Punkten eher Nachteile (obwohl bei 12 Punkten der Vergleich mit Signum schon unentschieden ausgeht), wenn auch die großen Lettern hervorragend sind. Außerdem muß man natürlich konstatieren, daß die jetzigen Vektorfonts Handarbeit sind und von denen großer Firmen wie Compugraphic oder Linotype auch eine verbesserte Qualität zu erwarten ist.

Ein anderes Problem, das meines Erachtens nicht sehr gut gelöst ist, ist die Definition von Stammseiten bzw. Stammelementen. Hier ist das Prinzip des PageMakers oder auch des Publishing-Partners, das feste Stammseiten verwendet, doch etwas komfortabler.

Ein großer Vorteil im Vergleich zu anderen Programmen auf dem ST ist die vielseitige Integration von Vektor- und Rastergrafiken. Vor allem mit Vektorgrafiken kann man seine Layouts durch Illustrationen verschönern, die weit bessere Qualität als Rasterbilder haben. Wenn dann erst die 'große' Version mit eigenem Vektor- und Business-Grafikeditor kommt...

Ein besonderer Vorzug ist der eingebaute Texteditor, über den bisher, wie ich glaube, noch kein anderes DTP-System verfügt.

Im allgemeinen gestaltet sich die Arbeit mit Calamus sehr komfortabel. Das Programm ist mächtig und die Einarbeitung nicht allzu langwierig und schwierig. Dennoch, in Sachen Komfort ist die älteste Konkurrenz die beste: Der gute alte Publishing-Partner löst viele Aufgaben eleganter, so z.B. Stammseiten und manuelles Unterschneiden. In anderen Punkten (Textedierung, Vielseitigkeit) hat Calamus die Nase vorn.

Auf's Papier

Nachdem Sie sich durch den Dschungel der Layoutwerkzeuge und -möglichkeiten gekämpft haben, wollen Sie natürlich auch ein Ergebnis auf dem Papier.

Im Augenblick unterstützt Calamus eine ganze Reihe von Druckern: zum Beispiel Epson FX-80, NEC P6/7, ATARI Laser, HP Laserjet+. Außerdem ist in Kürze eine Hardware-Lösung verfügbar, die den Anschluß an einen Linotronic Laser-Belichter möglich macht. Dies dürfte allerdings um die 2.500,-DM kosten. Die Druckereinstellbox bietet einige nützliche Optionen wie Vergrößerung, Druckreihenfolge, seitenverkehrten oder negativen Druck und Vergröberung des Druckrasters für Füllmuster. Letzteres verhindert oft zu schwarze Füllmuster oder Raster bei ungleichmäßiger Kontrastverteilung des Druckers.

Unverständlicherweise hat man nicht an eine Option für das getrennte Drucken gerader und ungerader Seitennummern gedacht. Wer also mit einem Laserdrukker Blätter beidseitig bedrucken will, muß es weiterhin auf die mühsame Art und Weise tun. Ärgerlich.

Über die Schriftqualität habe ich mich ja bereits ausgelassen, aber um es noch einmal kurz zu sagen: Große Schriften sind hervorragend, kleine Schriften bis 12 oder 10 Punkte sicherlich auch nicht nennenswert schlechter als SIGNUM-Schriften. Weniger gut ist die Schriftqualität bei den Mini-Größen, wenn man keinen Laser-Belichter verwenden will. Die Vektorgrafikqualität ist sehr gut, Rastergrafiken je nach Auflösung. Für den Druck der Rasterflächen erweist sich die Vergröberungsoption als sehr wichtig; erst bei der halben Rasterauflösung ist zumindest unser ATARI-Laser in der Lage, saubere Grauflächen zu drucken.

Das Handbuch

Meine Begeisterung über die Anleitung zu diesem nicht gerade kleinen Programm hält sich doch sehr in Grenzen. Zwar ist das Ganze übersichtlich, aber oft ungenau oder unklar. Außerdem fehlt ein Tutorium und, ganz besonders, ein Register. Zwar soll Anfang des Jahres ein Buch erscheinen (war schon für den Herbst geplant), das Anfängern helfen soll, aber dieses Buch muß man extra kaufen, was ich nicht ganz fair finden kann. Ein Programm für den Consumer-Markt sollte nicht eine derartige *'Nachkauf'*-Politik propagieren. Allerdings ist der Mangel DMC durchaus bewußt, man findet, wie man mir mitteilte, lediglich keine Zeit für ein Tutorium. Nun ja.

Noch viel ärger vermisse ich allerdings ein Register. Hoffentlich fehlt es nicht nur deshalb, weil die Index-Funktion von Calamus noch nicht alphabetisch sortieren kann...

Das Urteil

Was mir an Calamus sehr viel besser gefällt als an sämtlichen Konkurrenzprodukten, sind gar nicht einmal so sehr die jetzigen Fähigkeiten des Systems. Von diesen sind manche besser, manche schlechter als bei der Konkurrenz. Das Konzept von Calamus läßt aber immer erkennen, daß dieses Programm auf Erweiterungen und zusätzliche Fähigkeiten zugeschnitten ist, die der professionelle Anwender sich wünscht. Die Verwendung von Vektorfonts ist ein Beispiel, der modulare Aufbau ein anderes. Calamus macht einfach einen 'weltoffenen' Eindruck. Es ist ein leistungsfähiges, wenn auch (natürlich) noch nicht völlig ausgereiftes System. Auf jeden Fall ein guter Einstieg. Trotzdem, es gibt noch viele Schwächen und Fehler zu beseitigen, und man sollte genau prüfen, ob man, bis es soweit ist, mit Calamus arbeiten kann.

CS

Bezugsadresse:

Alle ATARI-Fachhändler

ENDE

Einkaufsführer

Hier finden Sie Ihren Atari Fachhändler

1000 Berlin



DATAPLAY
Bundesallee 25 · 1000 Berlin 31
Telefon: 030/861 91 61

Computare
Keithstr. 18-20 • 1000 Berlin 30
030/21 390 21
186 346 com d





... wir machen Spitzentechnologie preiswert.
Vertragshändler
UNION ZEISS
Kurfürstendamm 57 • 1000 Berlin 15
Telefon 32 30 61

1000 Berlin



Computershop
Behrendt, Reinecke, Tscheuschner GbR
Riesen Software-Angebot
Fürbringerstraße 26 · 1000 Berlin 61
Tel. (0 30) 6 91 76 66 · BTX (030) 6 91 76 66

COMPUTER-STUDIO
Schlichting
... die etwas andere Computerei
ATARI-Fachmarkt
MS-DOS Fachmarkt · NEC-Fachhandel
Katzbachstraße 8 · 1000 Berlin 61
030/7 86 43 40

2000 Hamburg

Bit Computer Shop
Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/49 44 00
Createam
Computer Hard & Software
Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon Sa. Nr. 0 40/641 50 91

Hardware
Software
Beratung
Service


ATARI Systemfachhändler
Münsterstraße 9 · 2000 Hamburg 54
Telefon 040/56 60 1-1



Computer & Zubehör-Shop
Gerhard u. Bernd Waller GbR
Kieler Straße 623
2000 Hamburg 54
040/570 60 07
BTX 040 570 52 75

RADIX Bürotechnik
Heinrich-Barth-Straße 13
2000 Hamburg 13
Telefon (0 40) 44 16 95
NEU: Software Shop

2000 Norderstedt



2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt
Heiligengeiststr. 20, 2120 Lüneburg
Tel. 0 41 31 / 4 61 22, Btx 40 24 22
Mo.-Fr. 9⁰⁰-18⁰⁰ und Sa. 9⁰⁰-13⁰⁰

2210 Itzehoe

Der Computerladen
Inhaber Ulrich Bubel · Martin Kopplow
Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90/91

2300 Kiel



Die Welt der Computer
Dreiecksplatz Nr. 7
2300 Kiel 1 · 04 31 / 56 70 42

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 5 60 57
BUF

2350 Neumünster



2390 Flensburg



2800 Bremen



Faulenstraße 48—52
2800 Bremen 1
Telefon (04 21) 17 05 77

2850 Bremerhaven

HEIM- UND PERSONALCOMPUTER



Kurt Neumann
„Bürger" 160
2850 Bremerhaven
Tel. 04 71 / 4 20 06

HARDWARE · SOFTWARE · PAPIERWARE

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler

Markstr. 52
2940 Wilhelmshaven
Telefon 0 44 21 - 2 61 45

3000 Hannover

COM DATA

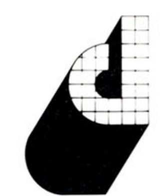
Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36



- Software
- Hardware
- Organisation
- Beratung
- Schulung

Großer Hillen 6
3000 Hannover 71
0511 - 52 27 11

3000 Hannover



DATALOGIC COMPUTERSYSTEME
ATARI ST- COMPUTER HARDWARE SOFTWARE
BERATUNG SERVICE VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL.: 0511 - 32 64 89

trendDATA Computer
IBM · EPSON · TRIUMPH ADLER
HEWLETT PACKARD · ATARI etc.

trendDATA Computer GmbH
Am Marstall 18–22 · 3000 Hannover 1
Telefon (05 11) 1 66 05-0

3040 Soltau

F & T Computervertrieb

Am Hornberg 1
(Industriegeb. Almhöhe)
3040 Soltau
Tel. 0 51 91 / 1 65 22

3150 Peine

Wieckenberg & Schrage GmbH
Computertechnik
Hard- u. Software

Woltorfer Str. 8, 3150 Peine
Tel. 0 51 71 / 60 52/3 o. 0 51 73 / 79 09

3170 Gifhorn

C OMPUTER
H AUS
G IFHORN

MITGLIED DER COMTEAM
DIE COMPUTER-PARTNER

INH. AXEL RITZ — POMMERNRING 38
D-3170 GIFHORN — TELEFON (0 53 71) 5 44 98
CELLER-BERLIN-BLUES MAILBOX — (0 51 41) 8 28 39

3300 Braunschweig

COMPUTER STUDIO
BRAUNSCHWEIG

Rebenring 49-50
3300 Braunschweig
Tel. (05 31) 33 32 77 / 78

3400 Göttingen



3400 Göttingen-Weende
Wagenstieg 14 – Tel. 05 51 / 38 57-0

3470 Höxter



3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler

Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (05 61) 70 00 00

4000 Düsseldorf

BERNSHAUS GmbH
Bürotechnik — Bürobedarf

Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81

HOCO
EDV ANLAGEN GMBH
Ellerstraße 155
4000 Düsseldorf 1
Telefon 02 11 / 78 52 13

Hard und Software
Werner Wohlfahrtstätter

Atari
Public Domain
Atari Spiele
Atari Anwender

Ladenlokal
Irenenstraße 76c
4000 Düsseldorf-Unterrath
Telefon (02 11) 42 98 76

4130 Moers



- Service-Center
- ATARI Fachhändler
- Hardware
- Software
- Erweiterungen

COP Computer Service GmbH
Essenberger Straße 2 H · 4130 Moers
Telefon (0 28 41) 2 35 85

5010 Bergheim



EDV-Beratung · Organisation
Programmierung · Home/Personal-Computer
Software · Zubehör · Fachliteratur

Zeppelinstr. 7 · 5010 Bergheim
Telefon 0 22 71 - 6 20 96

5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

5300 Bonn



Gesellschaft für Computer- und Communicationstechnologie mbH
Hardware · Software · EDV-Zubehör
Telefon 02 28/22 24 08
COCO GmbH · Schumannstraße 2 · 5300 Bonn 1

5457 Straßenhaus

DR. AUMANN GMBH
Computer-Systeme
Schulstr. 12
5457 Straßenhaus
Telefon 0 26 34 - 40 81/2

5500 Trier



Güterstraße 82 · 5500 Trier
☎ 06 51 / 20 97 10
Fordern Sie unsere Zubehör-Liste an!

5600 Wuppertal

Jung am Wall

Wall 31—33
5600 Wuppertal 1
Telefon 02 02/45 03 30

5600 Wuppertal

COMPUTER FINKE

COMPUTER finke

ATARI - SYSTEMFACHHÄNDLER

KIPDORF 22 · 5600 WUPPERTAL 1 · TEL. 0202 45 32 33

HARDWARE · SOFTWARE · ZUBEHÖR · SERVICE · SCHULUNGEN



... wir machen Spitzentechnologie preiswert.

MEGABYTE
Computer Vertriebs GmbH

Friedrich-Engels-Allee 162
5600 Wuppertal 2 (Barmen)
Telefon (02 02) 8 19 17

5630 Remscheid

COM SOFT
Nordstraße 57 · 5630 Remscheid
Telefon (0 21 91) 2 10 33

5650 Solingen

MegaTeam
Hardware — Software
Zubehör — Service

Rathausstraße 1-3 · 5650 Solingen 1
Telefon (02 12) 4 58 88

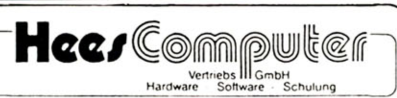
5800 Hagen



wir machen Spitzentechnologie preiswert.

Vertragshändler Axel Böckem
Computer + Textsysteme
Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Telefon (0 23 31) 7 34 90

5900 Siegen



Siegen · Weidenauer Str. 72 · ☎ 02 71/7 34 95

6000 Frankfurt

Müller & Nemecek

Kaiserstraße 44
6000 Frankfurt/M.
Tel. 0 69 - 23 25 44

WAIZENEGGER
Büroeinrichtungen

Kaiserstraße 41
6000 Frankfurt/Main
Tel. (0 69) 2 73 06 - 0

6000 Frankfurt



Eickmann Computer
Die Profis
Beratung, Service, Zubehör
In der Römerstadt 249
6000 Frankfurt/Main 90-Praunheim
Telefon (069) 76 34 09





Büro-Computer + Organisations GmbH

Oederweg 7-9
6000 Frankfurt/Main 1
☎ (0 69) 55 04 56 - 57

Commodore OKI ATARI TOSHIBA

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon (0 61 51) 5 60 57

6200 Wiesbaden

COMPUTER TREFF

Computerbedarf, PD und Software für ATARI, AMIGA, PC

Nettelbeckstraße 12
6200 Wiesbaden
Tel. (0 61 21) 40 43 02

6240 Königstein

KFC
COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6300 Gießen



6330 Wetzlar



Fachmarkt
für
Computer u.
Unterhaltungs-
electronic in Wetzlar,
Einkaufszentrum Bahnhofstraße, Tel. (0 64 41) 4 85 66

6400 Fulda

Schneider ATARI Commodore

WEINRICH

BÜRO · ORGANISATION

Ronsbachstraße 32 · 6400 Fulda

Telefon (06 61) 4 92 - 0

6457 Maintal

Landolt-Computer

Beratung · Service · Verkauf · Leasing

Wingertstr. 112
6457 Maintal/Dörnigheim
Telefon 0 61 81 - 4 52 93

6500 Mainz

:ELPHOTEC

Computer Systeme

Ihr Atari Systemhändler mit eigenem Service-Center

Walpodenstraße 10
6500 Mainz
Telefon 0 61 31 - 23 19 47

6520 Worms

ORION

Computersysteme GmbH
Friedrichstraße 22
6 5 2 0 W O R M S
Tel. 0 62 41 / 67 57 - 67 58

6700 Ludwigshafen

MKV Computermarkt

Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

THEILLE Computersysteme

Gilgenstraße 4 · 6720 Speyer
Telefon (0 62 32) 7 72 16

6800 Mannheim

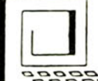


Computersysteme + Textsysteme

6800 Mannheim 24

Casterfeldstraße 74-76
☎ (06 21) 85 00 40 · Teletex 6 211 912

6800 Mannheim



Computer-Center
am Hauptbahnhof GmbH

L 14, 16-17
6800 Mannheim 1
Tel. (06 21) 2 09 83 / 84

6900 Heidelberg

JACOM FAMILA-CENTER

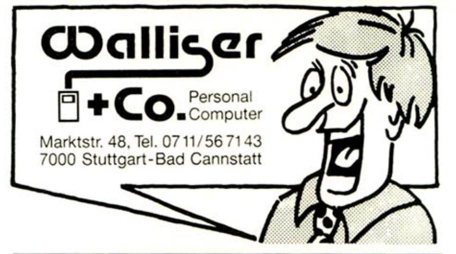
Hardware · Software
Schulung · Service

Hertzstraße 1 · 6900 Heidelberg 1
Telefon (0 62 21) 30 24 37

HEIDELBERGER COMPUTER CENTER

Bahnhofstraße 1
6900 Heidelberg
Telefon 06221/27132

7000 Stuttgart



7022 L-Echterdingen

Autorisierter ATARI-System-Fachhändler

ATARI ST

Matrai computer

Matrai Computer GmbH
Bernhauser Str 8
7022 L-Echterdingen
☎ (07 11) 79 70 49

7030 Böblingen

Verkauf – Service – Software

Norbert Hlawinka
Sindelfinger Allee 1
7030 Böblingen
Tel. 0 70 31 / 22 60 15



COMPUTER CENTER

7100 Heilbronn



7100 Heilbronn



Computer-Welt

Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 02

7101 Flein

der **COMPUTERLADEN** von



In der Falterstraße
7101 Flein

Beratung, Service, Verkauf,
Software-Entwicklung
direkt beim Peripherie-Hersteller

7150 Backnang

Computer-Fans finden bei uns alles von:



7400 Tübingen

Werner Brock

COMPUTERSTUDIO

Poststraße 2—4 · D-7400 Tübingen
Tel. (0 70 71) 3 43 48 · Fax (0 70 71) 3 47 92

Autorisierter Systemfachhändler für:
ATARI, Schneider, Commodore, Panasonic, Kaypro, Sharp, NEC, OKI, STAR...

7410 Reutlingen

Werner Brock

COMPUTERSHOP

Federnseestr. 17 · 7410 Reutlingen
Tel. 0 71 21 - 3 42 87
Tx 1 72 414 024 RMI D · box:rmi:taisoft · Fax 0 71 21 - 33 97 79

Autorisierter Systemfachhändler für:
ATARI, Schneider, Commodore, Panasonic, Kaypro, Sharp, NEC, OKI, STAR,...

7450 Hechtingen



Gesellschaft für Datenverarbeitung mbH

Computer · Drucker
Zubehör · Fachliteratur
Schloßplatz 3 · 7450 Hechingen
Telefon 0 74 71 / 1 45 07

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis

HEIM + PC-COMPUTERMARKT

HARDWARE · SOFTWARE · LITERATUR



ATARI COMMODORE CUMANA DATA-BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON

7475 Meßstetten 1 · Hauptstraße 10 · 0 74 31 / 6 12 80

7480 Sigmaringen

Wir stellen uns für Sie auf den Kopf!
Wir führen:
Hardware - Software - Zubehör
Zeitschriften - Bücher
Wir sind autorisierte Fachhändler von:
ATARI Schneider
Tandon
SOFT&EASY COMPUTER GMBH
7480 Sigmaringen Am Rappgässle Tel. 07571/12483
7968 Saulgau Pfarrstraße 13 Tel. 07581/2598

7500 Karlsruhe



MKV GMBH
Kriegsstraße 77
7500 Karlsruhe
Telefon (0721) 84613

7600 Offenburg

FRANK LEONHARDT ELECTRONIC
Ihr Fachgeschäft für Microcomputer · Hifi · Funk
In der Jeuch 3
7600 Offenburg
Telefon 0781/57974

7640 Kehl/Rhein



Computer-Software-Marketing
eigener Service · eigene Software
Badstrasse 12
Tel. 07807/822
Telex: 752913
7607 NEURIED 2
Filiale:
Hauptstrasse 44
Tel. 07851/1822
7640 KEHL/RHEIN
ELEKTRO-MÜNTZER GmbH

7700 Singen



Ringstraße 4
Telefon (07731) 68222

7730 VS-Schwenningen

BUS BRAUCH & SAUTER COMPUTER TECHNIK
Villinger Straße 85
7730 VS-Schwenningen
Telefon 07720/38071-72

7750 Konstanz

ATARI ★ PC's ★ SCHNEIDER
computer - fachgeschäft
rösler
Rheingutstr. 1 · ☎ 07531-21832

7800 Freiburg

PYRAMID COMPUTER GMBH
KARTÄUSERSTRASSE 59
D-7800 FREIBURG/BRST.
TELEFON 0761-382038

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag ☎ 06151/56057
BUF

7890 Waldshut-Tiengen

hettler-data
service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 07751/3094

7900 Ulm

HARD AND SOFT COMPUTER GMBH
Ulms großes Fachgeschäft
für BTX, Heim- u. Personalcomputer
Hafenbad 35 · 7900 Ulm/Donau
Telefon (0731) 62699

EDV-Systeme
Software-erstellung
Schulung
Systemhaus:
Frauenstraße 28
7900 Ulm/Donau
Tel. (0731) 28076
Telex 712973 csulm-d
COMPUTER STUDIO

7918 Illertissen

biotech gmbh
technische Informationsysteme
Computerladen
Marktplatz 13
7918 Illertissen
07303/5045

7980 Ravensburg

GRAHLE
Expert Grahle Computer
Eisenbahnstr. 33
7980 Ravensburg
Tel.: 0751/15955
Vertragshändler für ATARI, Schneider und Star

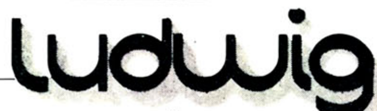
8000 München



star NEC
HDS-COMPUTER-VERTRIEBS GMBH
Am Klostergarten 1 · 8000 München 60
Tel. (089) 837063-64







COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STRASSE 62L
EURO-INDUSTRIE-PARK · 8000 MÜNCHEN 45
TELEFON 089/3113066 · TELETEX 898341

schulz computer
Schillerstraße 22
8000 München 2
Telefon (089) 597339
Beratung · Verkauf · Kundendienst

8032 Gräfelfing



COMPUTER SYSTEME SCHULUNG
Am Haag 5
8032 Gräfelfing
Tel. 089-8545464.851043

8100 Garmisch-Partenk.

Uwe Langheinrich Elektronik Center
Hindenburgerstr. 45
8100 Garmisch-Partenkirchen
Tel. 08821-71555
Bitte Gratisliste anfordern

8150 Holzkirchen

ATARI

Besuchen Sie uns!

Fordern Sie unseren Software-Katalog (520ST) an!

MÜNZENLOHER GMBH
Tölzer Straße 5
D-8150 Holzkirchen
Telefon: (0 80 24) 18 14

8170 Bad Tölz

Uwe Langheinrich Elektronik Center
Wachterstr. 3
8170 Bad Tölz
Tel. 0 80 41 - 4 15 65
Bitte Gratisliste anfordern

8330 Eggenfelden



Computer-Centrum
R. Lanfermann

Schellenbruckstraße 6
8330 Eggenfelden
Telefon 0 87 21 / 65 73

Altöttinger Straße 2
8265 Neuötting
Telefon 0 86 71 / 71 6 10

8400 Regensburg

Zimmermann elektroland

8400 Regensburg
Dr.-Gessler-Str. 8
☎ 09 41 / 9 50 85

8390 Passau
Kohlbruck 2a
☎ 08 51 / 5 20 07

8423 Abensberg

COMPUTERVERSAND WITTICH
Tulpenstr. 16 · 8423 Abensberg
☎ 0 94 43 / 453



8500 Nürnberg

EINE IDEE ANDERS
KARSTADT NÜRNBERG AN DER LORENZKIRCHE
TECHNIK CENTER
1. KLASSE EINKAUFEN IM WELTSTADTHAUS

hib
HIB Computer GmbH
Äußere Bayreuther Str. 57a – 59
8500 Nürnberg 10
Tel.: (09 11) 56 29 26 · Telex: 17 – 91 18 253
Teletex: 2627 – 91 18 253 · Telefax: (09 11) 51 30 40
Systemfachhändler für anspruchsvolle Computertechnik
ATARI · SONY · brother · Microsoft · TOSHIBA · Zenith data systems · EPSON

8520 Erlangen



Computerservice Decker
Meisenweg 29 - 8520 Erlangen
Telefon 0 91 31 / 4 20 76

Zimmermann elektroland

8520 Erlangen
Nürnberger Straße 88
Tel. (0 91 31) 3 45 68

8500 Nürnberg
Hauptmarkt 17
Tel. (09 11) 2 07 98

8600 Bamberg



8700 Würzburg



Hardware · Software
Service · Schulung

computer center
am Dominikanerplatz
Ruf (09 31) 3 08 08 - 0

8720 Schweinfurt

Uhlenhuth GmbH
Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 65 21 54

8900 Augsburg



Unser Plus: Beratung u. Service
Schwalbenstr. 1 · 8900 Augsburg-Pfersee
Telefon (08 21) 52 85 33 oder 52 80 87
Computer Vertriebs- und Software GmbH

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057 BUF

ÖSTERREICH

A-1040 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H
A-1040 Wien · Paniglgasse 18-20
Tel. (02 22) 5 05 78 08, 5 05 88 93

A-8010 Graz







Erfolgreich werben

Sprechen Sie mit uns.
Heim-Verlag ☎ (0 61 51) 5 60 57 BUF

INSERENTEN-VERZEICHNIS

Kleinanzeigen

BIETE HARDWARE

Computerbörsen: 26. + 27.11.: Neuss, 27.11: Köln, 3. + 4.12.: Wuppertal, 10. + 11.12.: Wattenscheid, 17. + 18.12.: Stolberg, 17. + 18.12!: Oberhausen Info: 02845-27260

ATARI 520 ST + 1MB, ROM TOS SM 124, SF 314 inkl. Software pcditto und ST Manager Text zu verkaufen. VB 1100,– DM Tel: 040-6476725 (ab 18 Uhr)

++++Hard- und Software++++ Z. B. HDplus 20 für DM 1145,– Lfw. 3,5-Zoll DM 325,– Info hl-Data, Lothar Ullmann Birkenstr. 18, 7332 Eislingen Tel. 07161/83406

Atari-ST-Blitter-ROM-TOS mit Fast-Loader 100 DM, 02630/7525

ST260, SM124, SF314, Uhr DM 600 Interface für AT-Tastatur DM 150 2MB-Erweiter. f. ST260 DM 700 D. Hink 02101/540421

Mega ST4, SLM 804, SH 205, noch Garantie. VB 6900,- **0251/862594**

Verk. Speichererw. ☎ 0431569216

Atari 520+, Blitter Tos, Fast ROM Floppy 720KB, Star Writer, Psion Chess usw. 750 DM, T. 089/355697

Plotter HP 7475, Neupr. DM 5500,- ca. 10 Betr.-Std. wg. AD-Plotter zu verk. DM 3900,- Tel. 089/771957

SM 124, neu, 6 Mon. Garantie, DM 365,- Tel. 06123/62669

ST520+ 1MB ROM-TOS, SF314, SM124, SH205, RTOS-Mod., +1 Bücher + 3 Jg. St-Comp. 40 Disk + Org. Lattice C 3.04 + Tempus + GFABas + Comp + Psion Chess nur kpl. 2600 DM. Lamar 06831-73910 nach 20 Uhr

260ST (1MB) **kompl.,** 2 Laufw. 20 MB Hardd., **ext. Tast.** (ST-Tast), 2400,–; + 9-Nadel-Drucker 2650,– Fr. Diekert, 0711/1211228 (9-18 h)

1040STF + SM124 = 950,-; Thomson Color Monitor = 550,-; Profiprinter von C-Itoh anschlußfertig = 500,-; alles Bestzustand; Komplettpreis = 1800,-; 0211-4557125

Farbmonitor Comm. 1084 (neu) 500,- Easytizer (Videodigitizer) 200,- Aladin 100,- Orig. Spiele: Giana Sis., Helloooon, Asterix, Road Runner je 30 DM Tel. 0431/311406

Verkaufe Atari 1040 ST + SM124 + Maus Tel. 05323-2534 od. 5424

Blitter-TOS o. altes TOS + TOS 1.4 NEU gleichz. in allen ST, umfangr. Anleit. K. Ratsch, Herner Str. 127, 4350 Recklinghausen

Atari 260ST + 1MB + Rom-Tos + SF354 + 2×720 KB + Vortex HD20 + SM124 + Lit. Adimens + GFA3.0 + 1st-Word + Gabi 9009 DM 2200,- Tel. 02525/712177

Atari SF354; 70 DM 0202/660422

Atari/Schneider/Festpl/Scanner w. unter Listenpreis. Lieferbar! z. B.: Mega ST2/SM 124: 2.487,- Zubeh. Firma Tel.: 02323/46362

BIETE SOFTWARE

Computerbörsen: 26. + 27.11.: Neuss, 27.11: Köln, 3. + 4.12.: Wuppertal, 10. + 11.12.: Wattenscheid, 17. + 18.12.: Stolberg, 17. + 18.12!: Oberhausen Info: 02845-27260

Lohnsteuerjahresausgleich + Einkommensteuer 87/88. 30 DM J. Höfer, Grunewald 2a 5272 Wipperfürth 02192/3368

Verk. **ST Pascal + V2.08** Original Dokumentation DM 160,– VB F. Gollesch, Karlstr. 4, 5110 Alsdorf Tel. 02404-64428

Gem Drewa+, GEM Paint, Gem Write, Gem Diary, ST Learn, CAD 3D, Jet, **zum halben Preis** – 0212/208613

xxx **fibuMAN V3.0** xxx
x e 375,- f 725,- m 910,- x
x V-Scheck incl. Vers. NN + 5,-x
x H. Großkreuz Brunnenmarkt 2x
x 8490 Cham Tel. 09971/1584 x
xxx nach 18 Uhr 09971/3380 xxx

4D-Welt.Prg: Darstellung 4-dimensionaler Objekte, die im 4D Raum beliebig gedreht und verschoben werden können, auf dem SW-Schirm des ST. **Ernste Warnung:** Nur bestellen, wenn Sie einer Erweiterung Ihres räumlichen Vorstellungsvermögens psychisch gewachsen sind! Für DM 49 NN oder voraus bei: C. Pahl, Seestr. 46, 8200 Rosenheim. Info gegen Rückporto!!

■ ■ **Public Domain** ■ ■
Software
für alle
Atari ST-Modelle ■ ■
Sie suchen aktuellste PD-Soft? Dann fordern Sie unseren neuen Gesamtkatalog an, in dem wir neue PD's direkt aus England und selbstverständlich aus der BR Deutschland vorstellen! Übrigens: Wir haben nun einen neuartigen ST-Club gegründet
Klaus Kohler
Don-Carlos-Str. 33 B
7 Stuttgart 80

PD-Software
Gratiskatalog bei Olaf Schwede Röntgenweg 9/1, 7050 Waiblingen

Biete 2nd-Word 39,- 0711/602124

PD-Software ab 2,- DM/Diskette von PD1 bis heute. Auch große Anzahl „eigener" PD's lieferbar. Tel. 02721/2432 von 9–22 Uhr

★-ST-★ Public Domain ★-ST-★
Kostenlose PD-Liste auf Disk
mit Gratis.PRG anfordern
E. G. Wolf · Hufnerstr. 112
- 2000 Hamburg 60 - 040/615906 -
★-ST-★ für Atari-ST ★-ST-★

★★**NL10BOLD**★★ fett/kursiv Star NL10/IBM-Cartr. Info geg. Freiumschlag Krägelin, Lauenburger Str. 23, 7500 Karlsruhe

★★Brandneue PD-Software★★
★selbstverständlich Virenfrei★
★Fast täglich neue Programme★
★direkt vom Autor, die es nur★
★ bei mir gibt! Riesiges Ange-★
★ bot auf über 60 Seiten! **Alle**★
★Programme sind einzeln aus-★
★wählbar!!! Der absolute Hit:★
★★ nur 1,5 Pf pro KByteE ★★
★Sie können **eigene** Disketten★
★senden oder auf **Markendisks**★
★bestellen!! Jedem PD-Katalog★
★ liegt ein **Extrablatt** mit den★
★allerneusten Programmen, die★
★ ständig erfaßt werden, bei!! ★
★**Neu**★ **Extraservice** ★ **Neu**★
★ Ich **scanne** und **digitalisiere**★
★ für Sie jedes schwarz/weiß ★
★★oder Videobild in Degas★★
★ Bearbeitung erfolgt selbst- ★
★verständlich **postwendend!!!!**★
★ **G R A T I S** katalog beim ★
★ ST profi-partner ★
Mönkhofer Weg 126, 24 Lübeck 1
★★ Telefon: 0451-505367 ★★

Timeworks DTP V1.12 nur 250,- plus NN. Tel. 05251/49317

■■■**PD nur 4,80 DM** incl.■■■
■ **FUJI**-Disk. Immer aktuell ■
■ ab 10 Stck. **portofrei** ■
■**FUJI**-Disketten ab **12,90 DM**■
■ **24-Bestellservice** ■
■■■Schneider 0281/61772■■■

Digitalisiere Skizzen, Fotos, Videos, etc! Biete auch Starke Utility-Software! Info bei: PSD/Hofbrede 5a/483 Gütersloh

Public-Domain-Blitzversand!
Riesenauswahl! Preise:
ab DM 4,– einseitige Disketten
ab DM 6,– doppelseitige
Inclusive Diskette!!!
Auch alle ST-Disks! Auf Doppeldisks beliebig kombinierbar!
Gratisliste anfordern bei:
A. Gauger Software
Buhlstraße 16a, 7505 Ettlingen
0 72 43 / 3 18 28
Bitte Computertyp angeben!!

Lohn-Einkommensteuer, Miet-Lastenzuschuß, Rentenber./Beamtenversorgung.
H-I-SOFTWARE
Niederfelderstraße 44
8072 Manching · Tel. 08459-1669

★★ Public Domain Software ★★
IBM u. komp. Disk nur 5,– DM
Atari ST Disk ab 3,– DM
Riesenauswahl ★★★ Superpreise
Katalog-Disk für je 1,80 DM
Graf & Schick · Hauptstraße 32 a
8542 Roth · Tel. 0 91 71 / 50 58 - 59

STAD, PC-ditto, FSII je 100 DM Nur Originale 02261/56644

GROSSE GRAFIKSAMMLUNG! ca. 2000 IMG-Grafiken (z. B. für Wordplus) DM 25,– 06302-3338

Wärmebedarf DIN4701 + K-Zahl ★ Heizflächenauslegung ★ Rohrnetz ★ Demodisk 2-seitig für DM 10 Vorkasse von J. Binder, Eichendorffstr. 15 · 5030 Hürth

Einkommen-/Lohnsteuer 1988
f. Atari ST, C64, C128
alle Einkünfte, Sonderausgaben, agw. Belastng., Berlin-Präf. Ausführliche Anleitung, **69 DM** + Versandk. Info 1,50, **H. Bohnenkamp Waldstr. 15 8520 Erlangen**

FLUSI der Segelflugsimulator: F-Schlepp, Streckenflugplanung, Streckenflüge mit Fotos, Auswertung, 3-D-Darstellung. Demo: 15,- Programm: 80,- F. Meyer, Im Mitteldorf 9, 3006 Burgwedel 1

ORIGINALPROGRAMME
Riesenauswahl an neuesten Programmen zu den günstigen – Gauger Software Preisen – ... die Gelegenheit!!
Gratisliste anfordern bei:
A. Gauger Software, Buhlstr. 16a
7505 Ettlingen, 07243/31828

Super Spiel 4 gewinnt auf Zeit! nur 20 DM für ST Mono auf Disk. Georg Munsch, Zeppelinstr. 21 5270 G'bach 02261-65973

FIBU 4.1 .... DM59,– ausger. FIBU in GEM m. Anleit. 60 Konten ca. 2400 Buchungen mon. E/Ü-Rechn. GuV Bilanz MwSt autom. ★ Tel. 09429/8402

$Turbo-PD-Versand M. Gruber$
$ – Versand in 48 Stunden $
$ – Alle ST-Disks auf Lager $
$ – Virenfrei/Verify/updated $
$ – Nur Markendisketten $
$ – 1DD-Disk 5,10 DM $
$ – 2DD-Disk 8,10 DM $
$ (frei kombinierbar!) $
$ – laufend Sonderangebote $
$ – Versand/Porto 4,00 DM $
$ – PD-Liste gegen 1,00 DM $
$ Am Weingert 27, D-8411 $
$ Pettendorf, Tel. 09409/2271 $

Software aller Art, auch PD's, u.a. aus ST Computer. Heißer Tip! Neue Zeitschrift: Public-Domain. Torsten Duffner, Software-Service Ritterstr. 6, 7833 Endingen, 07642/3875

Große Public-Domain-Spielesammlung, 50 Spiele auf 5 Disk (2dd), nur gg. SM 124. Info bei Torsten Duffner, Software-Service, Ritterstr. 6, 7833 Endingen, Tel. 07642/3875

★-ST-★ Public Domain ★-ST-★
S - Kostenlose Liste auf Disk - S
U mit Gratis.PRG anfordern U
P- Karin Büst · Rennbahnstr. 20 -P
E- 2 Hamburg 74 · 040/6514966 -E
R ★PD★ für Atari ST ★PD★ R

★★★★ PD-SERVICE ★★★★
★ Einzeldiskette SS **DM 4,00** ★
★Doppeldiskette DS **DM 6,00**★
★ beliebig kombinierbar) ★
★ Porto u. Verpack. DM 3,00 ★
★V.-Scheck o. NN (+DM 3,50)★
★N. Twardoch, Gröchteweg 22★
★ 4902 Bad Salzuflen 1 ★

Wärmebedarf DIN4701 + K-Zahl ★ Heizflächenauslegung ★ Rohrnetz ★ Demodisk 2-seitig für DM 10 Vorkasse von J. Binder, Eichendorffstr. 15 · 5030 Hürth

PD Software für ST und PCditto 260 Disks ST, 1000 Disks PC Liste anfordern! Ab 4,50 DM incl. Disk Tel. und BTX 0431/311406

BTX und DFÜ mit ATARI ST BTX-Manager, BTX in s/w ab 328,- BTX-Term ST, s/w und F ab 229,- Modems von 290 – 498 DM Infos anfordern. Tel. u. BTX 0431311406

PD-Grafik-Bilder, Art-Lib für Degas + Stad. Katalog: Frey, Rheinstr. 12A, 6538 Münster-Sarmsheim

Lohn- und Einkommensteuer '88 + Tabellen + Was wäre wenn Mandantenfähig, Berlin, **30 DM** Dipl. F. J. Höfer, Grunewald 2a 5272 Wipperfürth, 02192/3368

Public-Domain-Soft! 1 PD 4 DM, 2 PD 6 DM. Gratisinfo: Miersch Klingelholl 53, 5600 Wuppertal 2 DS-Probedisk (720 KB PDs) 5 DM

2 PDs von ST-Comp. auf 2-s. Markendisk DM 5,- + NN DM 5,- soft-STation T. 07195/53707

Software zu einmaligen Preisen
★ Wir räumen unser Lager! ★
Zwei Katalogdisks erhalten Sie gegen 5 DM in Briefmarken!
G. Köhler, Soft- und Hardware Mühlgasse 6, 6991 Igersheim

Original **Adimens** (140 DM) und **Superbase** (170 DM) 0511-8236209

NEU! Urlaubsabrechnung! für S/W-Monitor; Info gg. Rückp./ Demo 20,- / Original m. Handbuch 49,- + NN W. Günsche, Wenck- 38 6100 Darmstadt 06151/712618

★-ST-★ SOFT aus 2000 ★-ST-★
GFA BASIC Int. V2.02 DM 19,-
OMIKRON.Basic Int. V3.0 DM 19
Postfach 740162 · 2 Hamburg 74

★★ Public Domain Software ★★
IBM u. komp. Disk nur 5,– DM
Atari ST Disk ab 3,– DM
Riesenauswahl ★★★ Superpreise
Katalog-Disk gegen Rückporto
Graf & Schick · Hauptstraße 32 a
8542 Roth · Tel. 0 91 71 / 50 58 - 59

STEVE 3.08 DM 250,- B. Severloh Am Ochsenmkt. 32, 2390 Flensburg

SUCHE HARDWARE

Kaufe ST's (a. def.) ☎ 0431569216

Suche Festplatte min. 30 MB nur Vortex Mo – Fr 8 – 16 0621/655493

ATARI SC 1224 Tel.: 07348/6327

SUCHE SOFWARE

Suche FORTRAN Tel. 08621-7082

TAUSCH

★ **PD-TAUSCH** ★ Info-Disk & Liste gegen Disk + Rückporto! D. Metz; Bahnhofstr. 84 a, D-2153 Neu Wulmstorf

Mainstream – der vielseitige Atari-ST-User-Club! Info gegen Rückporto von: Kay-Uwe Berghof Roseggerstr. 5, 5600 Wuppertal 2

Suche Kontakt m. **1040-Anwender** z. **praktischen** Zusammen-Arbeit in Wiesbaden/Mainz T. 06121/20667

VERSCHIEDENES

Computerbörsen: 26. + 27.11.: Neuss, 27.11: Köln, 3. + 4.12.: Wuppertal, 10. + 11.12.: Wattenscheid, 17. + 18.12.: Stolberg, 17. + 18.12!: Oberhausen Info: 02845-27260

+++ Hard- und Software +++ z. B. HDplus 20 für DM 1145,– Lfw. 3.5-Zoll DM 325,– Info: hl-Dta, Lothar Ullmann Birkenstr. 18, 7332 Eislingen Tel. 07161/83406

Atari ST User Club ★ Infos gegen frankierten Rückumschlag ★ Infodiskette für 5,00 DM M. Ludwig, Kruppstr. 19, 6290 Weilburg

Virus-Falle Hardware 49,95 DM Eprom-Brenner für ST 249,00 DM Roßmöller Meckenheim 022955206

Atari-Laser: Suche GEM-Fonts f. Treiber Laserbrain. M. Engel Platenstr. 8, 8520 Erlangen

Exkl. CP6-Farbdrucke/512 Farben Ihre eigenen DEGAS-, NEO- od. ART-Graphiken. Info anfordern bei: Brigitte Holtmann Ritterstr. 47, 5600 Wuppertal 2

Welcher **Assembler-Profi** kann den TT-Befehl im Atari-Logo a. d. kleinen Font... T: 069-893965

Verkaufe Atari IC40Stf komplettes System + Drucker Star LC/0+ Disketten Preis n. V. Tel. 02333/3271

Mac-Emu. Aladin mit org. ROMS; dBaseII 150,-, 1st Word+ 100,-, Easy-Draw 120,-; Tel. 07181/4763

★★★★★★★★★★★★★
★ Suchen Vertriebspartner ★
★ für folgende Programme! ★
★ arztMAN, optikMAN, ★
★ videoMAN , botanikMAN ★
★ fakturMAN, Fahrschule ★
★ Nachkalkulation, Personal- ★
★ verwaltung und ★
★ Tankstellenabrechnung. ★
★Alle Programme mehrfach im★
★ Einsatz. ★
★ Dipl. Ing. (FH) Nebauer ★
★ + Baumgartner ★
★ 8391 Untergriesbach ★
★ Leizesberg, Hauptstr. 17 ★
★ Tel.: 08586/4532 ★
★ o. 0851/83740 ★
★★★★★★★★★★★★★

Programmier Praxis



In unserer Dezember-Programmierpraxis beschäftigen wir uns diesmal hauptsächlich mit dem beliebten GFA-BASIC, doch auch für die Assembler- und C-Freaks ist etwas dabei.

Interessant dürfte für viele die nachlauffreie Scroll-Routine sein, die sich - da in Assembler geschrieben - auch problemlos in andere Programmiersprachen wie C, Pascal, Modula etc. einbinden läßt. Unser Beispiel ist allerdings in GFA-BASIC, was allerdings der Sache keinen Abbruch tun sollte.

Bei dem nächsten Beitrag geht es nicht um hüllenlosen Urlaub in der Karibik sondern um eine kleine Hardcopy-Routine für 9-Nadel-Drucker. Mit ihr ist ein streifenloser Ausdruck von monochromen Bildern möglich.

Damit Sie noch schnell eine Datei für die noch zu besorgenden Weihnachtsgeschenke anlegen können, bieten wir mit *Schnelle Dateien* die wichtigsten Grundlagen zur Programmierung der eigenen Datenbank.

Wer jetzt einen Beitrag über Walt Disneys Minnie Maus erwartet, hat sich leider getäuscht. In dieser Programmierpraxis finden Sie nur eine Mini-Maus-Bibliothek mit Makros zur Mausprogramierung in Assembler.

Last not least kommen die C-Freunde zum Zug. Wie man Zeichensätze vergrößern kann, zeigt die Routine *Bchar*.

INHALT

Nachlauffreies Scrolling

Martin Fangmeyer

Bei dieser Routine handelt es sich um ein Assembler-Progrämmchen, das es erlaubt, auch aus Hochsprachen (BASIC, C, Pascal, Modula) ein absolut nachlauffreies Text-Scrolling zu realisieren. Um ein großes Maß an Flexibilität zu gewährleisten, teilt sich die Routine in drei verschiedene Teile ein.

Als erstes gibt es einen Programmteil, der einen ganzen Bildschirm (25 Zeilen) auf einmal ausgibt.

An dieser Stelle kommt natürlich kein Text-Scrolling zum Tragen, vielmehr macht sich hier die Geschwindigkeit der Routine beim Ausgeben von Strings bemerkbar. Ca. 0.1 Sekunde muß für einen ganzen Schirm einkalkuliert werden. Diese Zeit ist etwa um den Faktor 4.5 gegenüber der normalen Textausgabe über das Betriebssystem schneller. Des weiteren erledigen zwei andere Programmteile das Scrolling nach oben und unten. Auch hier sorgen die Routinen für einen richtigen Turbo-Effekt, und störendes Flimmern, wie etwa bei den Betriebssystem-Routinen, kann man lange suchen.

Nun aber zum Aufruf des Assembler-Bröckchens. Wie aus dem Beispiel-Programm zu ersehen, verwende ich die Routinen in Verbindung mit GFA-BASIC. Deshalb werde ich auch meine Erklärungen in der Syntax dieser Sprache halten. Um jedoch auch den Freunden anderer Sprachen die Möglichkeit zum Verständnis zu geben, möchte ich zunächst den hier verwendeten Befehl *"C:adr%(L:string%,W:mode%)"* erläutern. Mit seiner Hilfe lassen sich von GFA-BASIC aus Maschinen-Routinen aufrufen. Dazu wird in der Variablen *adr%* die Adresse der entsprechenden Routine abgelegt. In *string%* findet sich beim Aufruf der beiden Scroll-Prozeduren die Adresse des auszugebenden Strings wieder. Dieser String sollte normalerweise die Länge von 80 Zeichen = einer Bildschirmzeile besitzen. Um jedoch auch kürzere Strings ausgeben zu können, läßt sich das Ende des Strings mit einem Null-Byte markieren. Der Rest des Bildschirms hinter dem Null-Byte wird dann gelöscht.

Der Parameter *mode%* (Wortlänge) bestimmt dabei die aufzurufende Funktion. Bei einer negativen Zahl (-1) wird der Bildschirm nach unten verschoben und der String in der obersten Zeile dargestellt.

Es wird also nach oben gescrollt. Genau die umgekehrte Richtung, nämlich nach unten, nimmt die Routine, wenn *mode%* auf einen positiven Wert gesetzt wird. Der Schirm wird dann nämlich nach oben verschoben und der String erscheint am unteren Rand Ihres Monitors.

Wenn Sie nun einen ganzen Bildschirm darstellen möchten, müssen Sie zuerst eine Tabelle erstellen, in der die Adressen aller auszugebender Strings aufgeführt sind. Wie man das in GFA-BASIC erledigt, kann man sich in Listing 1 ansehen.

An die Scroll-Routine übergeben Sie dann nicht mehr die Adressen der einzelnen Strings, sondern die Adresse dieser Tabelle. Für die Länge der Strings gelten die gleichen Bedingungen wie bei den eigentlichen Scroll-Routinen. Wie Sie bei dem Aufruf der Routine schon sehen können, ist der Parameter *mode%* in diesem Falle auf Null zu setzen.

Es ist nicht nötig, vor dem Aufruf den Mauszeiger oder den Cursor auszuschalten. Diese Aufgaben erledigt die Routine ohne Ihr Zutun.

Nach diesen Erläuterungen zur Benutzung nun aber weiter zur Arbeitsweise.

Als erstes werden der Maus- und der Textcursor ausgeschaltet, danach die Adressen des *Font-Images* und des *logischen* Bildschirmes ermittelt. Das *Font-Image* enthält die Daten, aus denen sich die einzelnen Zeichen zusammensetzen. Ein Zeichen belegt dort 16 Byte. Da es im Zeichensatz des ATARI 256 verschiedene Zeichen des 8*16 Systemfonts gibt, ist diese Tabelle folglich 4096 Bytes lang. Die einzelnen Bytes des Zeichens liegen

Einige Benchmarks: (Zeiten in Sekunden)

Darstellen von 10 kompletten Bildschirm-Seiten

	Betriebssystem	SCROLL
	4.8	1.1

Scrollen in einem Text, 1000 Zeilen

Blitter-TOS, mit Blitter

	Betriebssystem	SCROLL
Nach oben	37.8	22.9
Nach unten	37.4	22.8

Blitter-TOS, ohne Blitter

	Betriebssystem	SCROLL
Nach oben	49.8	22.9
Nach unten	46.3	22.8

Benchmarks

dabei immer um 256 Bytes verschoben in diesem Image. Will man also z.B. ein "A" ausgeben (ASCII-Code 65), muß zur Adresse des Font-Images noch diese 65 addiert werden, und schon ergibt sich die Adresse der ersten Pixelzeile des Zeichens. Die weiteren Pixelzeilen folgen um je- weils 256 Bytes versetzt. Da direkt auf den Zeichensatz zugegriffen wird, ist es möglich, alle Zeichen aus dem Zeichensatz darzustellen. Somit lassen sich auch die Steuercodes kleiner als 32 in Form von Symbolen ausgeben.

Danach wird, in Abhängigkeit vom Parameter *mode%*, in die einzelnen Teile der Scroll-Routine verzweigt. Diese Programmteile rufen dann, eventuell nach dem Scrollen des Bildschirms, die Routine *"AUSGABE"* auf. Dort wird ein String auf den Bildschirm geschrieben. Schließlich werden dann der Maus- und Textcursor wieder sichtbar gemacht und die Routine verlassen.

Um das Prinzip besser zu verstehen und um diese Routinen in der Anwendung zu sehen, habe ich ein kleines Programm geschrieben, das ein Gerüst für einen Text-Editor darstellt (s. Listing 2).

```
1:  a$=""                                 ! a$ löschen
2:  FOR a%=anfang% to anfang%2            ! Zählschleife
3:   a$=a$+MKL$(VARPTR(feld$(a%)))        ! Tab.d.Adr.erstellen
4:  NEXT a%                               ! nächst. Feldelement
5:  tab%=VARPTR(a$)                       ! Adr.d.Tab.ermitteln
6:  a%=C:adr%(L:tab%,0)                   ! Routine aufrufen
```

Listing 1: Ermitteln der Adressen der auszugebenden Strings

```
1:   ' © 1988 Martin Fangmeyer
2:   '        Wilmeresch 60
3:   '        4430 Steinfurt 1
4:   '
5:   RESTORE scroll       ! Datazeiger setzen
6:   READ len%            ! Länge der Routine lesen
7:   scroll$=SPACE$(len%*4)   ! Speicher für die Routine
                               reservieren
8:   scroll%=VARPTR(scroll$)  ! Adr.d.Speichers holen
9:   FOR a%=1 TO len%
10:    READ wert%
11:    LPOKE scroll%,wert%    ! Routine i.d.Speicher
                               (L)poken
12:    ADD scroll%,4          ! Nächstes Langwort
13:  NEXT a%
14:  '
15:  DIM text$(1000)     ! Stringfeld dimensionieren
16:  FOR a%=0 TO 1000    ! Dieses Feld mit zufälligem
                           Inhalt füllen
17:    text$(a%)=LEFT$(STR$(a%)+"  "+STRING$(78,RANDOM
              (223)+32),80)
18:  NEXT a%
19:  '
20:  a$=SPACE$(100)      ! Platz für 25*4 Byte
                           (eine Bildschirmseite)
21:  adr%=VARPTR(a$)     ! Adresse des Strings ermitteln
22:  FOR a%=0 TO 24      ! Die Adressen der Text-Strings
                           festhalten
23:    LPOKE adr%+a%*4,VARPTR(text$(a%))
24:  NEXT a%
25:  '
26:  scroll%=VARPTR(scroll$)   ! Die Adresse der Scroll-
                                 Routine holen
27:  VOID C:scroll%(L:adr%,0)  ! Die erste Textseite
                                 ausgeben
28:  '
29:  PRINT CHR$(27);CHR$(101);   ! Cursor einschalten
30:  '
31:  REPEAT
32:    a%=INP(2)                   ! Auf Tastendruck warten
33:    IF a%=200                   ! Hoch (scrollen)
34:      DEC zeile%
35:      IF zeile%<0               ! Eventuell bei 1000
                                     neu anfangen
36:        zeile%=1000
37:      ENDIF
38:      IF CRSLIN=1               ! Scrollen notwendig
39:        scroll%=VARPTR(scroll$)
40:        VOID C:scroll%(L:VARPTR(text$(zeile%)),-1)
               ! Scroll-Routine aufrufen
41:      ELSE
42:        PRINT CHR$(27);CHR$(65); ! Andernfalls Cursor
                                      eins höher
43:      ENDIF
44:    ELSE
45:      IF a%=208             ! Runter (scrollen)
46:        INC zeile%
47:        IF zeile%>1000      ! Evtl. bei 0 neu anfangen
```

```
48:          CLR zeile%
49:        ENDIF
50:        IF CRSLIN=25       ! Scrollen notwendig
51:          scroll%=VARPTR(scroll$)
52:          VOID C:scroll%(L:VARPTR(text$(zeile%)),1)
                ! Scrollroutine aufrufen
53:        ELSE
54:          PRINT CHR$(27);CHR$(66); !Sonst nur Cursor
                                      höher
55:        ENDIF
56:      ELSE
57:        IF a%=20                    ! Links
58:          PRINT CHR$(27);CHR$(68); ! Cursor eine Pos.
                                       nach links
59:        ELSE
60:          IF a%=205                  ! Rechts
61:            PRINT CHR$(27);CHR$(67); ! Cursor eine
Pos.
                                         nach rechts
62:          ELSE          ! Taste ist kein Steuerzei-
chen.
63:            OUT 2,a%    ! Daher das Zeichen auf dem
                           Schirm ausgeben
64:          ENDIF
65:        ENDIF
66:      ENDIF
67:    ENDIF
68:  UNTIL a%=27            ! Solange, bis ESC gedrückt
wird
69:  `
70:  scroll:
71:  DATA 152
72:  DATA 809238555,1627390504,809238630,1627390496,
          -1610595334,37364392
73:  DATA -1400822,1060896771,1313756303,1140458028,
          578833007,551680
74:  DATA 9725696,18620480,-772014080,83914815,
          1289240318,1223180030
75:  DATA -87012136,1056852200,1056897744,1289240318,
          1223180030,-87012136
76:  DATA 1056852200,1056897744,1289240318,1223180030,
          -87012136,1056852200
77:  DATA 1056897744,1289240318,1223180030,-87012136,
          1056852200,1056897744
78:  DATA 1289240318,1223180030,-87012136,1056852200,
          1056897744,1372127130
79:  DATA 745472004,611975588,-704905216,2013275196,79,
          1627390136,1610613116
80:  DATA 541118972,30240,1883196632,1056852200,
          1056834768,1289240318
81:  DATA 1223180030,80760024,1056852200,1056834768,
          1289240318,1223180030
82:  DATA 80760024,1056852200,1056834768,1289240318,
          1223180030,80760024
83:  DATA 1056852200,1056834768,1289240318,1223180030,
          80760024,1056852200
84:  DATA 1056834768,1289240318,1223180030,80777724,960,
        1372127124
85:  DATA 745472004,611975440,607911936,5202218,
          1610612976,574357504
86:  DATA 1584128,544145412,608578648,607911936,5202190,
          -591659008
87:  DATA 83907017,-1286144,13247098,14041728,270420480,
          1734792128
88:  DATA 345052521,16777296,359203328,10491241,
          50331888,359203840
89:  DATA 20977001,83886480,359204352,31462761,
          117441072,359204864
90:  DATA 41948521,150995664,359205376,52434281,
          184550256,359205888
91:  DATA 62920041,218104848,359206400,73405801,
          251659440,1380602314
92:  DATA -7188875,1108492842,5259818,10502698,15745578,
          20988458,26231338
93:  DATA 31474218,36717098,41959978,47202858,52445738,
          57688618,62931498
94:  DATA 68174378,73417258,78664330,1372258238,
          1316306688,1060896770
95:  DATA 1060896771,1313692815,1316302908,1794538,
          809238629,1642373129
96:  DATA 1316290560,0,0,0,0
```

Listing 2: Kleines Grundgerüst für einen Text-Editor

```
 1:   MOVE #27,D0
 2:   BSR BCONOUT
 3:   MOVE #102,D0        ; Cursor ausschalten
 4:   BSR BCONOUT
 5:   DC.W $A000          ; LINE-A installieren
 6:   LEA CHARS(PC),A1    ; Umständlich, aber PC-relativ
 7:   MOVE.L -$16(A0),(A1) ; Adresse des Zeichensatzes
                            speichern
 8:   DC.W $A00A          ; Maus ausschalten
 9:
10:   MOVE #3,-(SP)
11:   TRAP #14
12:   ADDQ.L #2,SP
13:            ; In D0 findet man nun die Bildschirm-
14:                                        ; Adresse
15:   LEA LOGBASE(PC),A1
16:   MOVE.L D0,(A1)      ; Bildschirmadresse speichern
17:
18:   TST 8(SP)           ; Flag:
19:   BMI RAUF            ; Kleiner als 0: Raufscrollen
20:   BEQ SEITE           ; Gleich 0: Ganze Seite
                            darstellen
21:   MOVE.L D0,A0        ; Ansonsten: Runterscrollen
22:   ADD.L #1280,A0      ; 1. Zeile löschen
23:   MOVEQ.L #63,D0      ; 64 Durchläufe => 24 Zeilen
                                               bewegen
24:
25:  SCROLL_DOWN:
26:   MOVEM.L (a0)+,D1-D7/A1-A5 ; mit 2 Zeilen
                          werden 48 Byte kopiert
27:   MOVEM.L D1-D7/A1-A5,-1328(A0) ; 1
28:   MOVEM.L (a0)+,D1-D7/A1-A5
29:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 2
30:   MOVEM.L (a0)+,D1-D7/A1-A5
31:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 3
32:   MOVEM.L (a0)+,D1-D7/A1-A5
33:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 4
34:   MOVEM.L (a0)+,D1-D7/A1-A5
35:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 5
36:   MOVEM.L (a0)+,D1-D7/A1-A5
37:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 6
38:   MOVEM.L (a0)+,D1-D7/A1-A5
39:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 7
40:   MOVEM.L (a0)+,D1-D7/A1-A5
41:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 8
42:   MOVEM.L (a0)+,D1-D7/A1-A5
43:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 9
44:   MOVEM.L (a0)+,D1-D7/A1-A5
45:   MOVEM.L D1-D7/A1-A5,-1328(a0) ; 10
46:   DBRA D0,SCROLL_DOWN  ; insgesamt: 64*10*48=30720
                             Byte kopieren
47:
48:   MOVE.L 4(SP),A6          ; Adresse des Strings vom
                                 Stack holen
49:   MOVE.L LOGBASE(PC),A2    ; Bildschirmadresse in A2
50:   ADD.L #30720,A2          ; Nur unterste Zeile
                                 schreiben
51:   MOVE.L #79,D2            ; 80 Zeichen
52:   BSR AUSGABE              ; Zeile ausgeben
53:   BRA ZURÜCK ; Und zurück zum aufrufenden Programm
54:
55:  RAUF:                     ; Raufscrollen
56:   MOVE.L D0,A0             ; Bildschirmadresse in A0
57:   ADD.L #30240,A0          ; Unterste Zeile löschen
58:   MOVEQ #63,D0             ; 64 Durchläufe
                                 (0 zählt mit)
59:
60:  SCROLL_UP:
61:   MOVEM.L (A0)+,D1-D7/A1-A5
62:   MOVEM.L D1-D7/A1-A5,1232(A0)        ; 1
63:   MOVEM.L (A0)+,D1-D7/A1-A5
64:   MOVEM.L D1-D7/A1-A5,1232(A0)        ; 2
65:   MOVEM.L (A0)+,D1-D7/A1-A5
66:   MOVEM.L D1-D7/A1-A5,1232(A0)        ; 3
67:   MOVEM.L (A0)+,D1-D7/A1-A5
68:   MOVEM.L D1-D7/A1-A5,1232(A0)        ; 4
69:   MOVEM.L (A0)+,D1-D7/A1-A5
70:   MOVEM.L D1-D7/A1-A5,1232(A0)        ; 5
71:   MOVEM.L (A0)+,D1-D7/A1-A5
72:   MOVEM.L D1-D7/A1-A5,1232(A0)        ; 6
```

```
 73:    MOVEM.L (A0)+,D1-D7/A1-A5
 74:    MOVEM.L D1-D7/A1-A5,1232(A0)          ; 7
 75:    MOVEM.L (A0)+,D1-D7/A1-A5
 76:    MOVEM.L D1-D7/A1-A5,1232(A0)          ; 8
 77:    MOVEM.L (A0)+,D1-D7/A1-A5
 78:    MOVEM.L D1-D7/A1-A5,1232(A0)          ; 9
 79:    MOVEM.L (A0)+,D1-D7/A1-A5
 80:    MOVEM.L D1-D7/A1-A5,1232(A0)          ; 10
 81:    SUB.L #960,A0
 82:    DBRA D0,SCROLL_UP    ; 64*10*48=30720 Byte kop.
 83:
 84:    MOVE.L 4(SP),A6          ; Adresse des Strings
 85:    MOVE.L LOGBASE(PC),A2    ; Bildschirmadresse
 86:    MOVE.L #79,D2            ; 80 Zeichen
 87:    BSR.S AUSGABE            ; Oberste Zeile schreiben
 88:    BRA ZURÜCK       ; Zurück zum aufrufenden Programm
 89:
 90:   SEITE:            ; Eine ganze Seite schreiben
 91:    MOVE.L #24,D1    ; 25 Bildschirmzeilen
 92:    MOVE.L D0,D6     ; Bildschirmadr.in D6 speichern
 93:    MOVE.L 4(SP),A0   ; Adresse der Stringtabelle
                            25*4 Byte
 94:
 95:   NÄCHSTE_ZEILE:
 96:    MOVE.L D6,A2       ; Bildschirmadresse
 97:    MOVE.L (A0)+,A6    ; eine Zeile holen
 98:    MOVE.L #79,D2      ; 80 Buchstaben
 99:    BSR.S AUSGABE      ; eine Zeile ausgeben
100:    ADD.L #1280,D6     ; eine Cursorzeile tiefer
101:    DBRA D1,NÄCHSTE_ZEILE  ; nächste Zeile
102:    BRA ZURÜCK     ; Zurück zum aufrufenden Programm
103:
104:   AUSGABE:
105:    MOVE.L CHARS(PC),A1   ; Adr.d.Zeichensatzdaten
106:    CLR.L D0
107:    MOVE.B (A6)+,D0       ; Ein Byte holen
108:    TST.B D0              ; Byte testen
109:    BEQ.S ZEILE_ZU_ENDE   ; Zeile zu Ende (Byte = 0)?
110:    ADD.L D0,A1           ; + Offset
111:    MOVE.B (A1),(A2)      ; Zeichen auf den Bildschirm
                                bringen
112:    MOVE.B 256(A1),80(A2)
113:    MOVE.B 512(A1),160(A2)
114:    MOVE.B 768(A1),240(A2)
115:    MOVE.B 1024(A1),320(A2)
116:    MOVE.B 1280(A1),400(A2)
117:    MOVE.B 1536(A1),480(A2)
118:    MOVE.B 1792(A1),560(A2)
119:    MOVE.B 2048(A1),640(A2)
120:    MOVE.B 2304(A1),720(A2)
121:    MOVE.B 2560(A1),800(A2)
122:    MOVE.B 2816(A1),880(A2)
123:    MOVE.B 3072(A1),960(A2)
124:    MOVE.B 3328(A1),1040(A2)
125:    MOVE.B 3584(A1),1120(A2)
126:    MOVE.B 3840(A1),1200(A2)  ; 16 MOVE-Befehle
127:
128:   NÄCHSTES_ZEICHEN:
129:    ADDQ #1,A2        ; eine Zeichenposition weiter
130:    DBRA D2,AUSGABE   ; und nächstes Zeichen ausgeben
131:    RTS
132:
133:   ZEILE_ZU_ENDE:     ; Nach einem Null-Byte wird die
134:    CLR.B (A2)        ; Zeile bis zum rechten Rand
                            gelöscht
135:    CLR.B 80(A2)
136:    CLR.B 160(A2)
137:    CLR.B 240(A2)
138:    CLR.B 320(A2)
139:    CLR.B 400(A2)
140:    CLR.B 480(A2)
141:    CLR.B 560(A2)
142:    CLR.B 640(A2)
143:    CLR.B 720(A2)
144:    CLR.B 800(A2)
145:    CLR.B 880(A2)
146:    CLR.B 960(A2)
147:    CLR.B 1040(A2)
148:    CLR.B 1120(A2)
149:    CLR.B 1200(A2)
150:    ADDQ.L #1,A2
151:    DBRA D2,ZEILE_ZU_ENDE ; Weiter bis zum rechten
                                Rand
152:    RTS
153:
154:   BCONOUT:               ; ein Zeichen 'normal'
                                ausgeben, z.B.
155:                          ; um den Cursor ein- und
                                auszuschalten
156:    MOVE D0,-(SP)         ; Zeichen auf den Stack
157:    MOVE #2,-(SP)         ; 'Zielgerät' Bildschirm
                                        (Console)
158:    MOVE #3,-(SP)         ; Funktionsnummer
159:    TRAP #13              ; Bios aufrufen
160:    ADDQ.L #6,SP          ; Stack reparieren
161:    RTS
162:
163:   ZURÜCK:  ; Hier kehrt die Routine zu Basic zurück
164:    MOVE #27,D0       ; Escape-Sequenz einläuten
165:    BSR.S BCONOUT
166:    MOVE #101,D0      ; Cursor einschalten
167:    BSR.S BCONOUT
168:    DC.W $A009        ; Maus wieder einschalten
169:    RTS
170:
171:   CHARS:      DS.L 1   ; Speicher für Adresse des
                              Zeichensatzes
172:   LOGBASE:    DS.L 1   ; Speicher für Bildschirmadr.
173:
174:    END
```

Listing 3: Die Assembler-Routine

Hardcopy für 9-Nadler

Streifenlose Bräune/Schwärze

Michael Kraus

Der ST unterstützt von Natur aus 9 Nadel-Drucker. Die Anwender können zufrieden sein. 'Alternate+Help' gedrückt und schon erscheint der Bildschirminhalt auf dem Drucker. Doch nach näherem Hinsehen erkennt man, daß die Qualität nicht so ist, wie gewünscht.

Die Hardcopyfunktion, über 'Alternate+Help' erreichbar, hat die Angewohnheit, die 640 horizontalen Punkte des Bildschirms über 960 Druckpunkte auszudrucken.

Ein Punkt des Bildschirms entspricht also manchmal einem, manchmal zwei Druckpunkten auf dem Papier, was zu deutlich sichtbaren Schönheitsfehlern führt. Dem kann man Abhilfe schaffen, indem man den Grafikmodus, sprich die Punktdichte pro Inch, so ändert, daß auf 640 Bildpunkte tatsächlich 640 Dots auf das Papier kommen. Das hat nun aber den Nachteil, daß zum Beispiel eine gerade Linie recht deutlich als eine Aneinanderreihung von Punkten und nicht, wie es sein sollte, als gleichmäßiger Strich sichtbar wird. Außerdem ist solch eine Hardcopy recht blaß. Das kann durch zweimaliges Ausdrucken zwar ausgeglichen werden, strapaziert aber Geduld, Druckkopf und Ohr. Nicht zuletzt sind Hardcopies in DIN A4-Breite oft zu groß, um sie beispielsweise in Texte einzubinden.

Das vorliegende Programm versucht nun, diese Nachteile weitgehend auszugleichen. Zur Funktion und Arbeitsweise:

Das Bild, welches im gängigen Screenformat (32000 Bytes) abgespeichert sein muß, kann in zwei verschiedenen Größen gedruckt werden:

*Modus 1 mit ca. 68*47 mm*
*Modus 2 mit ca. 135*96 mm*

Um eine Verkleinerung des Bildes zu erreichen, wird wohlgemerkt nicht etwa jedes zweite Bit übersprungen o.ä., sondern die Punktdichte in beiden Richtungen vergrößert. Es kommen so nach wie vor 640*400 Bildpunkte aufs Blatt, es geht also keine "Information" verloren. Ein Punkt auf dem Bildschirm entspricht einem Dot auf dem Papier. Durch die hohe Punktdichte erreicht man außerdem "echte" Linien und eine hohe Kontrastwirkung.

Modus 1

Benutzt wird hier die vierfache Dichte horizontal (240 Punkte/Inch) und ein Punktabstand von 1/216 Inch vertikal (216 Punkte/Inch). Da der physikalische Abstand der Nadeln mit 1/72 Inch dreimal so groß ist, kann nur jede dritte Bitreihe des Bildschirms auf die acht Nadeln "gelegt" werden, danach rückt das Papier um 1/216 Inch nach oben und der Vorgang wiederholt sich noch zweimal. Jetzt sind also 3*8=24 Bitreihen ausgedruckt. Dann geht's von vorne los, bis der Bildschirm abgegrast ist.

Modus 2

Nun wird der Punktabstand verdoppelt: 120 Punkte/Inch horizontal und 108 Punkte/Inch vertikal. Jetzt können pro Durchlauf nur noch vier Nadeln belegt werden, da der Punktabstand von 1/108 Inch (vertikal) nur noch mit jeder zweiten Nadel zur Deckung kommt. Dies wird ebenfalls zweimal wiederholt, danach kommen die anderen vier Nadeln an die Reihe, um eine gleichmäßige Belastung zu erreichen. Mit Ausnahme von ein paar zusätzlichen Schleifen ist dieser Teil fast identisch mit dem von Modus 1. Schaut man mal genauer hin beim Drucken, erkennt man die Arbeitsweise recht leicht.

Das Programm ist in GFA-Basic geschrieben und mit einem Star NL10 betrieben worden, dessen Steuercodes identisch mit denen der Epson-9-Nadler sind.

PP

```
 1: ' =======================================
 2: ' ==  Hardcopies in Miniaturausführung ==
 3: ' ==  für Epson-kompatible Drucker     ==
 4: ' ==  von Michael Kraus, Schweinfurt   ==
 5: ' =======================================
 6: On Error Gosub Fehler  !Fehlerabfang
 7: Alert 3,"Miniatur-Hardcopy-Programm für|
          Epson-Drucker (und Kompatible)|

          Modus 1:  68 * 47 mm|  Modus 2: 135 * 96
          mm",0,"Modus 1|Modus 2",M%
 8: Fileselect "\*.*","",N$  !File waehlen
 9: If N$=""
10:   End
11: Endif
12: Bload N$,Xbios(2)         !File auf Bildschirm
13: If M%=1
14:   Gosub C1
15: Else
16:   Gosub C2
17: Endif
18: Run
19: Procedure C1                     !Modus 1
20:   Lprint Chr$(27);"3";Chr$(1);   !Abstand 1/216 inch
21:   For I%=0 To 399 Step 24
22:     For J%=0 To 2
23:       S1%=I%+J%
24:       S2%=S1%+3
25:       S3%=S2%+3
26:       S4%=S3%+3
27:       S5%=S4%+3
28:       S6%=S5%+3
29:       S7%=S6%+3
30:       S8%=S7%+3
31:       Lprint
Chr$(27);"*";Chr$(3);Chr$(128);Chr$(2);
                             !640 Bytes zu uebergeben
32:       For K%=0 To 639
33:         Bm%=128*Point(K%,S1%)+64*Point(K%,S2%)+32
              *Point(K%,S3%)+16*Point(K%,S4%)
34:         Bm%=Bm%+8*Point(K%,S5%)+4*Point(K%,S6%)+2
              *Point(K%,S7%)+Point(K%,S8%)
35:         Lprint Chr$(Bm%);    !Die Bytes drucken
36:       Next K%
37:       Lprint                 !1/216 inch weiter
38:     Next J%
39:     Lprint Chr$(27);"J";Chr$(20) !Weiter fuer
                                naechsten Durchgang
40:   Next I%
41: Return
42: Procedure C2     !Modus 2
43:   Lprint Chr$(27);"3";Chr$(2);   !Abstand 2/216 inch
44:   For H%=0 To 399 Step 24
45:     For I%=0 To 1
46:       B1%=2^(7-I%)
47:       B2%=2^(5-I%)
48:       B3%=2^(3-I%)
49:       B4%=2^(1-I%)
50:       For J%=0 To 2
51:         S1%=H%+I%*12+J%
52:         S2%=S1%+3
53:         S3%=S2%+3
54:         S4%=S3%+3
55:         Lprint Chr$(27);"*";Chr$(1);Chr$(128);
              Chr$(2);  !640 Bytes zu uebergeben
56:         For K%=0 To 639
57:           Bm%=B1%*Point(K%,S1%)+B2%*Point(K%,S2%)
                +B3%*Point(K%,S3%)+B4%*Point(K%,S4%)
58:           Lprint Chr$(Bm%);  !Die Bytes drucken
59:         Next K%
60:         Lprint               !2/216 inch weiter
61:       Next J%
62:       Lprint Chr$(27);"J";Chr$(14) !Weiter zum
                                Nadelwechsel
63:     Next I%
64:     Lprint Chr$(27);"J";Chr$(3)  !Weiter fuer
                                naechsten Durchgang
65:   Next H%
66: Return
67: Procedure Fehler             !Neustart bei Fehler
68:   Run
69: Return
```

Listing der Minihardcopy

ENDE

Schnelle Dateien

Franz Schott

Wer sich mit der Programmierung von Dateien beschäftigt, sollte die verschiedenen Lade- und Speicherzeiten berücksichtigen, die in den Tabellen in Bild 1 und 2 angegeben sind. Nutzen wir bei der Dateiverwaltung erstens die Schnelligkeit unseres ST und zweitens die des GFA-BASIC.

Wie aus der Tabelle in Bild 2 ersichtlich, werden zum Laden einer sequenziellen Datei mit Input# 104,4 Sekunden benötigt, während dieselben Datensätze mit dem durch >Block-R< gekennzeichneten Verfahren schon nach 5.71 Sekunden zur Verfügung stehen.

Es lohnt sich also, sich bereits bei der Planung eines Programmes hierüber einige Gedanken zu machen.

Die Tabellen in den Bildern 1 und 2 sind Hardcopies von dem in Listing 1 gezeigten Testprogramm (siehe nächste Seite). Bei Benutzung einer RAM-Disk ergeben sich weitere Vorteile, die bei einer Hard-Disk ähnlich liegen dürfte (ich konnte dies leider nicht testen).

Nun zu einzelnen Varianten: Gespeichert wurden 375 Datensätze zu je 75 Zeichen. Die unterschiedliche Länge der Dateien resultiert aus dem beim *PRINT#*-Befehl angehängten *Carriage Return (13)* und *Line Feed (10)*, das sind je Datensatz 12 Bytes. *PRINT#* und *INPUT#* sind sehr zeitaufwendig und sollten vor allem bei größeren Dateien nicht benutzt werden. Schneller geht es mit den *Random-Access-Dateien*, deren einzelne Felder und damit die Datensätze gleiche Längen haben - in unserem Beispiel 6 Felder mit 15, 10, 20, 8, 12 und 10 Zeichen (gesamt 75). Ein weiterer Vorteil dieser Dateiart mit *PUT#* und *GET#* ist die Zugriffsmöglichkeit auf einzelne Datensätze in der Datei.

Es geht aber Dank der leistungsfähigen GFA-Befehle noch schneller. Die gesamte Datei wird als Block von Diskette in einen String geladen bzw. aus einem String als Block gespeichert. Vor dem Laden muß natürlich ein String mit der erforderlichen Länge zur Aufnahme bereitstehen. Von diesem String (*Dat$* im Listing) aus werden dann die einzelnen Datensätze D$(374,5) aufbereitet. Beim Speichern werden die einzelnen Datensätze vor dem Speichern in den String aufgenommen. Das hört sich alles ein bißchen umständlich an, aber es ist die schnellste Methode. Die Datensätze können nach Herzenslust verändert, gelöscht und angefügt werden und sind dann schnell wieder abgespeichert. Man hüte sich jedoch davor, den String nach dem Motto *'Dat$=Dat$+D$(x,y)'* aufzubauen. Beim Ausprobieren bitte Kaffeepause einlegen. Im Testprogramm werden hierfür *BMOVE* und *Mid$* verwendet, in den Tabellen 1 und 2 ist diese Variante mit *Bsave* und *Block-R* gekennzeichnet.

Bleibt noch das Verfahren *Block-I* zu erklären. Es ist anzuwenden zum Laden von Dateien, die mit *PRINT#* abgespeichert sind. Die Datei wird zunächst ebenfalls als Block in einen vorbereiteten String geladen, dann aber wegen der unterschiedlichen Datenlängen Byte für Byte nach den abschließenden *Carriage Returns* abgesucht und so die einzelnen Datenfelder herausgefischt. Gegenüber dem *INPUT#*-Befehl ergibt sich eine Zeitersparnis von ca. 60%, es können alle sequenziellen Dateien damit eingelesen werden und evtl. nach vorheriger Aufbereitung auf einheitliche Längen mit *Bsave* wieder abgespeichert werden.

Nachdem nun die Vorzüge einer solchen Block-Datei aufgezeigt wurden, soll auch der Nachteil nicht verschwiegen werden. Die in den Tabellen dargestellte Dateilänge hat fast die maximale Größe von 32767 Bytes erreicht. Größer kann ein String nicht definiert werden (oder vielleicht doch mit dem neuen GFA V 3.0?).

Aber lassen wir den Kopf nicht hängen, unser GFA-BASIC hält ja noch weitere sehr gute Befehle bereit.

Wir *RESERVE*ieren uns einfach den benötigten Speicherbereich und verwenden statt *Varptr(Dat$) = HIMEM*.

Ein Test mit 2000 Datensätzen (150000 Bytes) ergab:

Speichern mit PUT#	179.17 Sekunden
Bsave (Himem)	48.25 Sekunden
Laden mit GET#	65.48 Sekunden
Bload (Himem)	35.06 Sekunden

Beim Reservieren von Speicher sollte man darauf achten, daß nicht mehr als

notwendig reserviert wird, andererseits die Datei aber auch in den reservierten Bereich hineinpaßt. Um beim Testen nicht bei jedem Start neuen Speicherplatz zu reservieren, kann der *RESERVE*-Befehl in eine If-Abfrage verpackt werden z.B. beim ST 1040:

```
If Himem >700000
    Reserve Fre(0)-300000
Endif
```

Nun noch einige Erklärungen zum Listing: Das Testprogramm besteht im Wesentlichen aus der Initialisierungsroutine mit dem Aufbau der Menüleiste und der Tabellengrafik sowie der Dimensionierung der Variablen, der Hauptschleife und den einzelnen Prozeduren.

Die benutzerfreundliche und einfach zu programmierende Menüleiste dürfte auch für den Einsteiger keine Probleme aufwerfen. Nach der Initialisierung wartet der ST in der nur 3 Zeilen großen Warteschleife auf einen Menüaufruf, landet dann in der Prozedur *Auswahl*, sucht sich hier den angeklickten Menüpunkt, arbeitet dann das entsprechende Unterprogramm ab und begibt sich, nachdem die Menüleiste aufgebaut ist, wieder zur Warteschleife.

Die einzelnen Prozeduren für das Laden und Speichern der Dateien bedürfen keiner besonderen Erklärung, die besonderen Funktionen wurden bereits erläutert. In Prozedur *Fil* werden Dateiname *(Fi$)* und Länge der Datei (*L%*, falls schon vorhanden) ermittelt. Die Prozedur *Zeit* errechnet den benötigten Zeitaufwand einschließlich Datenaufbereitung und gibt diese in einer Tabelle auf den Bildschirm aus. Mit der Prozedur *Ausgabe* werden die Datensätze auf dem Bildschirm ausgegeben, damit auch die letzten Zweifel auf das sichere Funktionieren beseitigt werden. Die Datensätze können mit der Maus vorwärts und rückwärts gescrollt werden. Durch Drücken beider Maustasten gelangt man zurück zur Warteschleife. Die umständliche Ausführung der *Print*-Befehle rührt von der ungünstigen Länge der Datensätze her. Die bereits festgelegten Feldlängen wollte ich nicht mehr ändern und die einzelnen Felder trotzdem mit einem Leerzeichen listen. Schließlich werden in der Prozedur *Anlegen* die Test-Datensätze erzeugt.

Es ist noch anzumerken, daß mit Rücksicht auf die Länge dieses Testprogrammes keine vollständige Fehlerbehandlung integriert ist. Diese darf in einem Anwendungsprogramm jedoch keineswegs fehlen. So muß z.B. das Programm abgefangen werden, wenn versucht wird, eine Datei abzuspeichern, ohne daß sich eine solche im Speicher befindet oder die Fehlermeldung nach *Gosub Fil* bei Anklicken von *Abbruch* und *ok* mit Leerstring (nur *Len(Fi$)>2* erlauben).

Diese Fehler können bei den einzelnen Prozeduren lokalisiert und bequem in eine gemeinsame Stringvariable verpackt werden, die dann am Ende der Prozedur *Auswahl* auf den Bildschirm ausgegeben werden kann und so auf unterlassene Bedienungshandlungen aufmerksam macht.



Lade- und Speicherzeiten von Dateien mit GfA-Basic in Sekunden

Länge Byte	SPEICHERN Print#	Put#	Bsave	Laden Input#	Get#	Block-R	Block-I	Datei
28125		3.58			3.78	3.12		\RAMPUT
28125			3.63		3.78	2.86		\RAMBSAVE
32625	7.69			91.38			36.24	\RAMPRINT

Bild 1: Speicherung auf RAM-DISK

Lade- und Speicherzeiten von Dateien mit GfA-Basic in Sekunden

Länge Byte	SPEICHERN Print#	Put#	Bsave	Laden Input#	Get#	Block-R	Block-I	Datei
28125		35.76			12.56	6.84		\DISKPUT
28125			11.98		12.62	5.71		\DISKSAVE
32625	46.82			184.48			39.97	\DISKPRT

Bild 2: Speicherug auf Diskette

```
 1:  ' ##################################
 2:  ' # Schnelle Dateien in GFA-BASIC #
 3:  ' #          Schott Franz          #
 4:  ' #       Bahnhofsplatz 3          #
 5:  ' #     8425 Neustadt/Donau        #
 6:  ' #       Tel. 09445/8176          #
 7:  ' ##################################
 8:  Gosub I.nit
 9:  Do
10:    On Menu
11:  Loop
12:  Procedure Auswahl
13:    Menu Off
14:    M0%=Menu(0)
15:    Cls
16:    If Wahl$(M0%)=" erstellen "
17:      Gosub Anlegen
18:    Endif
19:    If Wahl$(M0%)=" Datei "
20:      Gosub Ausgabe
21:    Endif
22:    If Wahl$(M0%)=" Zeittabelle "
23:      Sput Bild$
24:      Repeat
25:      Until Mousek Or Inkey$<>""
26:    Endif
27:    If Wahl$(M0%)=" Print# "
28:      Gosub S.ave_print
29:    Endif
30:    If Wahl$(M0%)=" Bsave "
31:      Gosub S.ave_bsave
32:    Endif
33:    If Wahl$(M0%)=" Put# "
34:      Gosub S.ave_put
35:    Endif
36:    If Wahl$(M0%)=" Get# "
37:      Gosub L.oad_get
38:    Endif
```

```
 39:     If Wahl$(M0%)=" Input# "
 40:       Gosub L.oad_input
 41:     Endif
 42:     If Wahl$(M0%)=" Block-R "
 43:       Gosub L.oad_block_r
 44:     Endif
 45:     If Wahl$(M0%)=" Block-I "
 46:       Gosub L.oad_block_i
 47:     Endif
 48:     If Wahl$(M0%)=" Ende "
 49:       Edit
 50:     Endif
 51:     Cls
 52:     Menu Wahl$()
 53:   Return
 54:   Procedure I.nit
 55:     Cls
 56:     Re$=Chr$(27)+"p"
 57:     Ra$=Chr$(27)+"q"
 58:     Print At(7,5);Re$'
 59:     Print "Lade- und Speicherzeiten von Dateien mit
 60:            GFA-BASIC in Sekunden ";Ra$
 61:     Print
 62:     Print '"Länge    S P E I C H E R N   L a d e n "
 63:     Print '"Byte     Print#    Put#    Bsave    Input#
 64:            Get#    Block-R    ";
 65:     Print "Block-I  Datei"
 66:     Line 0,50,639,50
 67:     Line 0,134,639,134
 68:     Line 0,94,639,94
 69:     Line 64,94,64,300
 70:     For N=132 To 630 Step 72
 71:       Line N,112,N,300
 72:     Next N
 73:     Line 276,94,276,300
 74:     Line 564,94,564,300
 75:     Sget Bild$
 76:     Cls
 77:     Dim D$(2500,5)
 78:     Dim Wahl$(35)
 79:     Dim F$(5)
 80:     Restore M.datas
 81:     I=0
 82:     Do
 83:       Read Wahl$(I)
 84:       Exit If Wahl$(I)="~~~"
 85:       Inc I
 86:     Loop
 87:     Wahl$(I)=""
 88:     M.datas:
 89:     Data DESKTOP, Information,————————
 90:     Data  , , , , , ,""
 91:     Data DATENSÄTZE, erstellen ,""
 92:     Data LADEN, Input# , Get# , Block-R , Block-I ,""
 93:     Data SPEICHERN, Print# , Put# , Bsave ,""
 94:     Data AUSGABE, Datei , Zeittabelle ,""
 95:     Data QUIT, Ende ,""
 96:     Data ~~~,""
 97:     On Menu  Gosub Auswahl
 98:     On Break Gosub Ende
 99:     Menu Wahl$()
100:   Return
101:   Procedure L.oad_get
102:     Gosub Fil
103:     T=Timer
104:     Open "R",#1,Fi$,75
105:     Field #1,15 As D0$,10 As D1$,20 As D2$,8 As D3$,
106:             12 As D4$,10 As D5$
107:     For N%=1 To L%/75
108:       Get #1,N%
109:       D$(N%-1,0)=D0$
110:       D$(N%-1,1)=D1$
111:       D$(N%-1,2)=D2$
112:       D$(N%-1,3)=D3$
113:       D$(N%-1,4)=D4$
114:       D$(N%-1,5)=D5$
115:     Next N%
116:     Z%=N%-1
117:     Close #1
118:     T1=Timer
119:     Sp%=47
120:     Gosub Zeit
121:   Return
122:   Procedure L.oad_block_r
123:     Gosub Fil
124:     If L%<=32767
```

```
125:       T=Timer
126:       Dat$=Space$(L%)
127:       Bload Fi$,Varptr(Dat$)
128:       Z%=0
129:       Pa%=1
130:       Feld:
131:       Data 15,10,20,8,12,10
132:       Repeat
133:         Restore Feld
134:         For R%=0 To 5
135:           Read F%
136:           D$(Z%,R%)=Mid$(Dat$,Pa%,F%)
137:           Add Pa%,F%
138:         Next R%
139:         Inc Z%
140:       Until Pa%=>L%
141:       T1=Timer
142:       Sp%=56
143:       Gosub Zeit
144:     Else
145:       Alert 3,"Datei ist zu groß.|Bitte Himem
146:                verwenden!",1,"klar",A%
147:     Endif
148:   Return
149:   Procedure L.oad_block_i
150:     Gosub Fil
151:     If L%<32767
152:       T=Timer
153:       Dat$=Space$(L%)
154:       Bload Fi$,Varptr(Dat$)
155:       Z%=0
156:       Pa%=0
157:       Do
158:         For R%=0 To 5
159:           Pe%=0
160:           Repeat
161:             Inc Pe%
162:           Until Mid$(Dat$,Pa%+Pe%,1)=Chr$(13)
                       Or Pe%>500
163:           D$(Z%,R%)=Space$(Pe%-1)
164:           Bmove Varptr(Dat$)+Pa%,Varptr(D$(Z%,R%)),
                                                          Pe%-1
165:           Add Pa%,Pe%+1
166:         Next R%
167:         Inc Z%
168:         Exit If Pa%=>L% Or Pe%>500
169:       Loop
170:       T1=Timer
171:       Sp%=65
172:       Gosub Zeit
173:     Endif
174:   Return
175:   Procedure L.oad_input
176:     L%=0
177:     Gosub Fil
178:     Print At(26,12);"B i t t e   G e d u l d"
179:     T=Timer
180:     If L%
181:       Z%=0
182:       Open "I",#1,Fi$
183:       Repeat
184:         For R%=0 To 5
185:           Input #1,D$(Z%,R%)
186:           Exit If Eof(#1)=-1
187:         Next R%
188:         Inc Z%
189:       Until Eof(#1)=-1
190:       Close #1
191:       T1=Timer
192:       Sp%=38
193:       Gosub Zeit
194:     Endif
195:   Return
196:   Procedure S.ave_put
197:     Gosub Fil
198:     L%=Z%*75
199:     T=Timer
200:     Open "R",#1,Fi$,75
201:     Field #1,15 As D0$,10 As D1$,20 As D2$,8 As D3$,
202:             12 As D4$,10 As D5$
203:     For N%=0 To Z%-1
204:       D0$=D$(N%,0)
205:       D1$=D$(N%,1)
206:       D2$=D$(N%,2)
207:       D3$=D$(N%,3)
208:       D4$=D$(N%,4)
```

```
209:       D5$=D$(N%,5)
210:       Put #1,N%+1
211:     Next N%
212:     Close #1
213:     T1=Timer
214:     Sp%=20
215:     Gosub Zeit
216:   Return
217:   Procedure S.ave_print
218:     Gosub Fil
219:     L%=Z%*87
220:     If Len(Fi$)>2
221:       T=Timer
222:       Open "O",#1,Fi$
223:       For N%=0 To Z%-1
224:         For R%=0 To 5
225:           Print #1,D$(N%,R%)
226:         Next R%
227:       Next N%
228:       Close #1
229:       T1=Timer
230:       Sp%=11
231:       Gosub Zeit
232:     Else
233:       Alert 2,"Keine gültiger Dateiname",1,"0k",A%
234:     Endif
235:   Return
236:   Procedure S.ave_bsave
237:     If Z%<436
238:       Gosub Fil
239:       L%=Z%*75
240:       If Len(Fi$)>2
241:         T=Timer
242:         Pa%=0
243:         Dat$=Space$(L%)
244:         For N%=0 To Z%-1
245:           Restore Feld
246:           For R%=0 To 5
247:             Read Pe%
248:             Bmove  Varptr(D$(N%,R%)),Varptr(Dat$)
                        +Pa%,Pe%
249:             Add Pa%,Pe%
250:           Next R%
251:         Next N%
252:         Bsave Fi$,Varptr(Dat$),L%
253:         T1=Timer
254:         Sp%=29
255:         Gosub Zeit
256:       Else
257:         Alert 2,"Keine gültiger Dateiname",1,"0k",A%
258:       Endif
259:     Else
260:       A$="Datei ist mit "+Str$(Z%*75)+" Bytes|zu
groP
261:           für den reservierten|Speicher"
262:       Alert 3,A$,1,"ok",A%
263:     Endif
264:   Return
265:   Procedure Fil
266:     Fileselect "\*.*","",Fi$
267:     If Exist(Fi$)
268:       Open "I",#1,Fi$
269:       L%=Lof(#1)
270:       Close #1
271:     Endif
272:   Return
273:   Procedure Zeit
274:     Sek=(T1-T)/200
275:     N%=0
276:     Ze%=0
277:     Repeat
278:       Inc N%
279:       If F$(N%)=""
280:         F$(N%)=Fi$
281:         Ze%=9+N%
282:       Endif
283:       If F$(N%)=Fi$
284:         Ze%=N%+9
285:       Endif
286:     Until Ze%>8
287:     Sput Bild$
288:     Print At(Sp%,Ze%);
289:     Print Using "###.##",Sek
290:     Print At(1,Ze%);
291:     Print Using "######",L%
292:     Print At(72,Ze%);Fi$
293:     Print At(30,24);Re$;" T A S T E ";Ra$'
294:     Sget Bild$
295:     Repeat
296:       If Inkey$="1"
297:         Bsave "zeittest",Varptr(Bild$),16000
298:       Endif
299:     Until Inkey$="a" Or Mousek
300:   Return
301:   Procedure Ausgabe
302:     N%=-1
303:     Repeat
304:       Inc N%
305:       For R%=0 To 5
306:         Print D$(N%,R%)'
307:       Next R%
308:       Print Chr$(27);"A"
309:     Until N%=23
310:     Do
311:       Mouse X,Y,K
312:       If K=1 And N%<Z%-1
313:         Inc N%
314:         Print At(1,25);
315:         For R%=0 To 5
316:           Print D$(N%,R%)'
317:         Next R%
318:         Print Chr$(27);"A"
319:       Endif
320:       If K=2 And N%>23
321:         Bmove Xbios(2),Xbios(2)+1280,30720
322:         Dec N%
323:         Print At(1,1);
324:         For R%=0 To 5
325:           Print D$(N%-23,R%);Chr$(0+(32 And R%<5));
326:         Next R%
327:       Endif
328:       Exit If K=3
329:     Loop
330:   Return
331:   Procedure Anlegen
332:     Print
333:     Print 'Re$;" Wieviele Datensätze sollen erstellt
334:         werden? ";Ra$'
335:     Input " ",Z%
336:     L%=(Z%)*75
337:     Print At(30,12);"Erstelle Datensatz Nr. "
338:     For N%=0 To Z%-1
339:       Print At(53,12);N%'
340:       D$(N%,0)="Datensatz"+Right$("...."+Str$(N%),5)
341:       D$(N%,1)=String$(10,Random(26)+65)
342:       D$(N%,2)=">20-Byte-Datenfeld< "
343:       D$(N%,3)=String$(8,Random(10)+48)
344:       D$(N%,4)=Left$(String$(6,Str$(N%)+"-"),12)
345:       D$(N%,5)="SchluPfeld"
346:     Next N%
347:   Return
```

Das Listing zu "Schnelle Dateien in GFA-BASIC"

Mini-Maus-Bibliothek

Ralph Egelseer

Eines der Probleme, unter denen Assembler-Programmierer oft leiden, ist die Abfrage der derzeitigen Position der Maus. Einen Lösungsvorschlag stellt die folgende Mini-Maus-Bibliothek für den GST Macro-Assembler dar, die sicher auch leicht auf andere Assembler übertragen werden kann. Sie stellt dem Programmierer eine Anzahl nützlicher Makros zur Beeinflussung und Abfrage des Mauszeigers zur Verfügung. Die Bibliothek kann durch einen am Anfang des Quelltextes plazierten INCLUDE-Befehl nachgeladen werden.

Folgende Funktionen stehen dann zur Verfügung:

1. *GEM_INIT:*
Diese Funktion muß am Anfang jedes Programmes stehen, das die Funktionen aus der Maus-Bibliothek benutzt. Beim Aufruf werden einige für GEM-Funktionen nötige Variablen (siehe unten) initialisiert.

2. *GEM_EXIT:*
Sollte vor Beendigung des Programms aufgerufen werden. Dadurch wird es beim GEM abgemeldet. Weitere GEM-Aufrufe sind danach nicht mehr möglich.

3. *MOUSEON:*
Mittels einer Funktion des Line-A-Emulators wird der Mauszeiger eingeschaltet.

4. *MOUSEOFF:*
Bevor man innerhalb eines Programmes Texte oder Grafiken auf dem Bildschirm ausgeben läßt, sollte man mit dieser Funktion den Mauszeiger ausschalten, da er sonst diese Ausgaben wieder zerstören kann.

5. *DEFMOUSE form;*

Mit dieser Funktion des AES kann der Mauszeiger auf eine von acht vordefinierten Formen geschaltet werden. Der Parameter *form* kann alle Adressierungsarten des M68000 annehmen, es muß aber in Wortgröße darauf zugegriffen werden können (gerade Adresse im Speicher!). Dabei gilt für *form*:

0: Pfeil
1: Schreibmarke
2: Biene
3: zeigende Hand
4: flache Hand
5: dünnes Fadenkreuz
6: dickes Fadenkreuz
7: hohles Fadenkreuz

6. *GETMOUSE x,y,buttons*

Dieser Macro gibt nach Aufruf einer VDI-Funktion die Bildschirmkoordinaten und den Status der Maustasten in den Parametern *x*, *y* und *buttons* zurück. Diese Parameter können alle Adressierungsarten, bis auf *"immediate"* (z.B. #34) annehmen. Auch auf sie muß wieder in Wortgröße zugegriffen werden können.

Der Quelltext, in den die Maus-Bibliothek eingebunden werden soll, muß folgende Bedingungen erfüllen:

1. Er darf keine der unten genannten Variablen enthalten.
2. Die Programmsektionen müssen die korrekten Namen TEXT, DATA und BSS tragen, da *GEM_INIT* zwischen diesen Sektionen umschaltet.

Das Demoprogramm testet alle diese Macros. Es gibt auf Druck irgendeiner Maustaste aus, welche es war, und an welcher Position sich der Mauszeiger befindet.

Verwendete Variablen in der Mausbibliothek:

AESPB bzw. *VDIPB*:
Anfangsadressen von Listen, die die Adressen von Variablenarrays enthalten, die für AES- bzw. VDI-Aufrufe benötigt werden.

CONTROL,GLOBAL, INT_IN,INT_OUT, ADDR_IN, ADDR_OUT:
Arrays, in denen die Parameterübergabe an bzw. von AES und VDI stattfindet. ADDR_IN und ADDR_OUT entsprechen bei VDI- Aufrufen den Arrays POINTS_IN bzw. POINTS_OUT. Die beiden eigentlich völlig verschiedenen Felder wurden zusammengelegt, um etwas Speicherplatz zu sparen.

VDIHANDLE:
Zwischenspeicher für die Nummer des Grafiktreibers. Diese Nummer wird für alle VDI-Aufrufe benötigt.

Verwendete Variablen im Demoprogramm:

X,Y:
Speicherstellen für die Koordinaten der Maus.

BUTTON1,BUTTON2:
Zwischenspeicher für den Status der Maus.

MAUSFORM:
Zähler für den Parameter, der die Form der Maus bestimmt.

DEZPUF:
Puffer, in dem die Ziffern, die das Unterprogramm DEZOUT erzeugt, vor der Ausgabe abgelegt werden.



```
  1: ****************************************************
  2: *                                                  *
  3: * Makrobibliothek für Mausprogrammierung           *
  4: *                                                  *
  5: * geschrieben 1988 von Ralph Egelseer              *
  6: *                                                  *
  7: ****************************************************
  8:
  9: ;--------------------------------------------------
 10: ;- Initialisierung des GEM
 11: ;--------------------------------------------------
 12:
 13: gem_init  MACRO
 14:
 15:           SECTION  data
 16: aespb     DC.L     control,global,int_in,int_out,
                        addr_in,addr_out
 17: vdipb     DC.L     control,int_in,addr_in,int_out,
                        addr_out
 18:
 19:           SECTION  bss
 20: control   DS.W     12
 21: global    DS.W     15
 22: int_in    DS.W     128
 23: int_out   DS.W     128
 24: addr_in   DS.W     128
 25: addr_out  DS.W     128
 26: vdihandle DC.W     1
 27:
 28:           SECTION  text
 29:           MOVE.W   #10,control        ;Appl_Init
 30:           CLR.W    control+2
 31:           MOVE.W   #1,control+4
 32:           CLR.L    control+6
 33:           MOVE.L   #aespb,D1
 34:           MOVE.W   #200,D0
 35:           TRAP     #2
 36:
 37:           MOVE.W   int_out,D0   ;Bei Fehler:
                                 Programm abbrechen
 38:           CMP.W    #-1,D0
 39:           BNE      geminit1
 40:           CLR.W    -(SP)
 41:           TRAP     #1
 42:
 43: geminit1  MOVE.W   #77,control        ;Graf_Handle
 44:           CLR.W    control+2
 45:           MOVE.W   #5,control+4
 46:           CLR.L    control+6
 47:           MOVE.L   #aespb,D1
 48:           MOVE.W   #200,D0
 49:           TRAP     #2
 50:           MOVE.W   int_out,vdihandle;Handleretten

 51:
 52:           ENDM
 53:
 54: ;--------------------------------------------------
 55: ;- verlassen des GEM
 56: ;--------------------------------------------------
 57:
 58: gem_exit  MACRO
 59:
 60:           MOVE.W   #19,control        ;Appl_Exit
 61:           CLR.W    control+2
 62:           MOVE.W   #1,control+4
 63:           CLR.L    control+6
 64:           MOVE.L   #aespb,D1
 65:           MOVE.W   #200,D0
 66:           TRAP     #2
 67:
 68:           ENDM
 69:
 70: ;--------------------------------------------------
 71: ;- Mauszeiger einschalten
 72: ;--------------------------------------------------
 73:
 74: mouseon   MACRO
 75:
 76:           DC.W     $A009
 77:
 78:           ENDM
 79:
 80: ;--------------------------------------------------
 81: ;- Mauszeiger ausschalten
 82: ;--------------------------------------------------
 83:
 84: mouseoff  MACRO
 85:
 86:           DC.W     $A00A
 87:
 88:           ENDM
 89:
 90:
 91: ;--------------------------------------------------
 92: ;- Form des Mauszeigers bestimmen; >form< v. bis 7
 93: ;--------------------------------------------------
 94:
 95: defmouse  MACRO    form
 96:
 97:           MOVE.W   #78,control        ;graf_mouse
 98:           MOVE.W   #1,control+2
 99:           MOVE.W   #1,control+4
100:           MOVE.W   #1,control+6
101:           CLR.W    control+8
102:           MOVE.W   [form],int_in
103:           MOVE.L   #aespb,D1
104:           MOVE.W   #200,D0
105:           TRAP     #2
106:
107:           ENDM
108:
109: ;--------------------------------------------------
110: ;- Holt die Koordinaten der Maus nach >mousex< u.
111: ;- >mousey<, den   Status nach >mousebut<
112: ;--------------------------------------------------
113:
114: getmouse  MACRO    mousex,mousey,mousebut
115:
116:           MOVE.W   #124,control       ;vq_mouse
117:           CLR.W    control+2
118:           CLR.W    control+6
119:           MOVE.W   vdihandle,control+12
120:           MOVE.L   #vdipb,D1
121:           MOVE.W   #115,D0
122:           TRAP     #2
123:
124:           MOVE.W   addr_out,[mousex]
125:           MOVE.W   addr_out+2,[mousey]
126:           MOVE.W   int_out,[mousebut]
127:
128:           ENDM
```

Listing 1: Makrobibliothek für die Mausprogrammierung

```
  1: ****************************************************
  2: *                                                  *
  3: *   Demoprogramm für die Mausbibliothek            *
  4: *                                                  *
  5: *   geschrieben 1988 von Ralph Egelseer            *
  6: *                                                  *
  7: ****************************************************
  8:
  9:           INCLUDE  maus.mac
 10:
 11:           SECTION  text
 12:
 13:           gem_init
 14:
 15:           mouseoff
 16:
 17:           PEA      clrtext ; Bildschirm löschen
 18:           MOVE.W   #9,-(SP)
 19:           TRAP     #1
 20:           ADDQ.L   #6,SP
 21:
 22:           PEA      starttext ; Startmeldung
 23:                              ausgeben
 24:           MOVE.W   #9,-(SP)
 25:           TRAP     #1
 26:           ADDQ.L   #6,SP
 27:           CLR.W    mausform
 28: loop1
 29:           mouseon
 30:           defmouse mausform
 31: loop2
 32:           getmouse x,y,button1 ; Warten bis Taste
                                    gedrückt
 34:           TST.W    button1
```

```
35:            BEQ.S     loop2
36:  loop3
37:            getmouse  x,y,button2; Warten bis Taste
38:                                 losgelassen
39:            TST.W     button2
40:            BNE.S     loop3
41:
42:            mouseoff
43:
44:            PEA       xtext   ; X-Koordinate
45:                              ausgeben
46:            MOVE.W    #9,-(SP)
47:            TRAP      #1
48:            ADDQ.L    #6,SP
49:            MOVE.W    x,D0
50:            BSR       dezout
51:
52:            PEA       ytext   ; Y-Koordinate
53:                              ausgeben
54:            MOVE.W    #9,-(SP)
55:            TRAP      #1
56:            ADDQ.L    #6,SP
57:            MOVE.W    y,D0
58:            BSR       dezout
59:
60:            CMP.W     #1,button1 ; Linke Taste
61:                                 gedrückt?
62:            BNE       lab1
63:            PEA       lttext     ; Ja: entspr.
64:                                 Text ausgeben
65:            MOVE.W    #9,-(SP)
66:            TRAP      #1
67:            ADDQ.L    #6,SP
68:            BRA       lab2
69:
70:  lab1      PEA       rttext     ; Text für rechte
71:                                 Taste ausgeben
72:            MOVE.W    #9,-(SP)
73:            TRAP      #1
74:            ADDQ.L    #6,SP
75:
76:  lab2
77:            PEA       tasttext   ; Ergänzenden
78:                                 Text ausgeben
79:            MOVE.W    #9,-(SP)
80:            TRAP      #1
81:            ADDQ.L    #6,SP
82:
83:            ADDQ.W    #1,mausform ; Mausform
84:                                  weiterschalten
85:            ANDI.W    #7,mausform
86:
87:            MOVE.W    #7,-(SP)    ; Tastendruck
88:                                  abwarten
89:            TRAP      #1
90:            ADDQ.L    #2,SP
91:            CMP.W     #'q',D0     ; Q-Taste
92:                                  gedrückt?
93:            BNE       loop1       ; Nein: Nochmal
94:                                  das Ganze
95:
96:            mouseon
97:            gem_exit
98:            CLR.W     -(SP)   ; Zurück zum Desktop
99:            TRAP      #1
100:
101: ; Ausgabe der unteren 16 Bit von D0 als Dezimalzahl
102:
103: dezout    LEA.L     dezpuf,A6
104:
105: dloop1    AND.L     #$FFFF,D0
106:           DIVU      #10,D0
107:           SWAP      D0
108:           ADDI.W    #48,D0
109:           MOVE.B    D0,(A6)+
110:           SWAP      D0
111:           TST.W     D0
112:           BNE       dloop1
113:
114: dloop2    MOVE.B    -(A6),D0
115:           MOVE.W    D0,-(SP)
116:           MOVE.W    #2,-(SP)
117:           TRAP      #1
118:           ADDQ.L    #4,SP
119:           CMP.L     #dezpuf,A6
120:           BHI.S     dloop2
121:
122:           RTS
123:
124:           SECTION   data
125:
126: clrtext   DC.B      27,'E',0
127: starttext DC.B      'Demoprogramm für Maus-Makro-
128:                      Bibliothek',13,10,10,0
129: xtext     DC.B      'X = ',0
130: ytext     DC.B      ' ; y = ',0
131: tasttext  DC.B      'gedrückt.',13,10,10,0
132: lttext    DC.B      ' ; Linke Taste ',0
133: rttext    DC.B      ' ; Rechte Taste ',0
134:
135:           SECTION   bss
136:
137: x         DS.W      1
138: y         DS.W      1
139: button1   DS.W      1
140: button2   DS.W      1
141: mausform  DS.W      1
142: dezpuf    DS.B      5
143:
```

Listing 2: Demoprogramm für die Mausbibliothek

Zeichensätze vergrößern

Franz Melchior

Bei der Entwicklung von GEM hat man viele Möglichkeiten der Gestaltung geschaffen, auch eher überflüssige. Andererseits aber hat man sich bei einigen Funktionen mit mittelmäßigen Ergebnissen zufriedengegeben: Eine maximale Textgröße von 26 Bildpunkten mag zwar bei niedriger Auflösung recht eindrucksvoll sein, reicht aber bei hoher Auflösung keineswegs zur wirkungsvollen Anwendung.

Meine Ansicht war es, eine Funktion in " C" zu programmieren, die diese "Not" lindert. Ernsthafte Programmierer werden das für eine überflüssige Spielerei halten, ich möchte aber doch behaupten, daß es auch dafür ernsthafte Anwendungen gibt!

Zuerst brauchte ich die Adresse der Systemfonts. Hier kam mir der Line-A-Befehl mit dem Opcode "0xa000" gelegen. Er liefert im Register A1 die Adresse einer Tabelle mit den Adressen der drei Systemfonts (6x6, 8x8 und 8x16). Daß er nur im Assemblercode erreichbar ist, stellt bei MEGAMAX kein Problem dar (siehe Funktion *'fontad()'*).

An der erhaltenen Adresse befindet sich im Betriebssystem jeweils ein Datenblock im VDI-Fontformat. Es wäre also elegant gewesen, ein *'struct'* zu definieren und die einzelnen Komponenten so anzusprechen. Da dies den Aufwand aber unnötig erhöht hätte, und die Funktion ja immerhin in einer 'Library' untergebracht werden, also kurz sein, sollte, entschloß ich mich, die Werte über Adreßdistanzen zu bekommen.

Distanz von
76: Zeiger auf die Fontdaten
82: Anzahl der Scanlinien = Höhe der Zeichen-"Zelle"

Jetzt galt es, die verfügbaren Informationen in entsprechender Vergrößerung auf den Bildschirm zu kopieren. Dazu besorge ich mir ein Byte der Fontdaten als Muster und übertrage jedes Bit je nach Vergrößerung mehrmals auf ein Byte des (physischen) Bildschirmspeichers, siehe Bild auf der nächsten Seite.

Diese Arbeit übernimmt die Funktion *'bchar()'* ("b" für big). Auf diesem Weg sind theoretisch Vergrößerungen von 1 bis unendlich möglich, sinnvoll jedoch nur bis etwa 25. Die Funktion erwartet als Parameter:

1. den auszugebenden Buchstaben
2. die x-Koordinate
3. die y-Koordinate
4. ein "Modusbyte", das die Vergrößerung und die Art der Ausgabe wie folgt enthält:

Bit 0 - Bit 4: Vergrößerung (1-31)
Bit 5: gesetzt -> transparent
nicht gesetzt -> überschreibend
Bit 6: gesetzt -> 8x8-Font
nicht gesetzt -> 8x16-Font
Bit 7: gesetzt -> invers
nicht gesetzt -> normal

Durch diese Aufschlüsselung ergibt sich folgende Möglichkeit der Definition von 'mode':

Bsp.:

13	13-fach vergrößerte 8x16- Fonts
13|128	wie oben, invertiert
13|64	wie oben, nur 8x8-Fonts
7|128|64	7-fach vergrößerte, inverse 8x8-Fonts

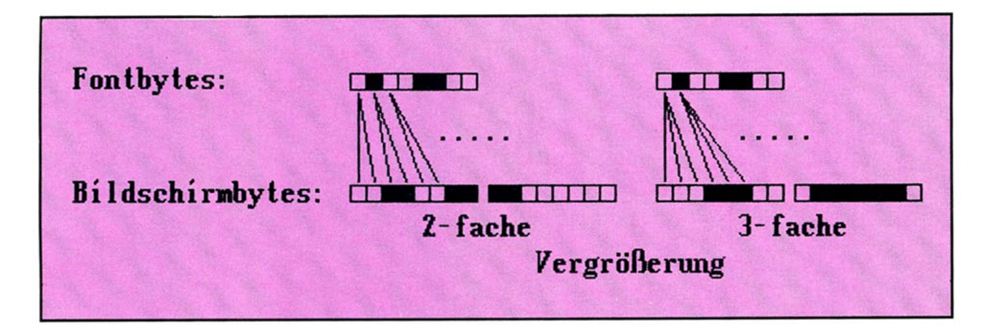
....

Um die Verwendung der Funktion zu erleichtern, gibt diese die Zeichenbreite zurück und ermöglicht dadurch die Plazierung der nachfolgenden Zeichen im richtigen Abstand (siehe Funktion *'bstring'*).

'bstring' übergibt *'bchar'* einzeln die auszugebenden Zeichen des Strings und erhöht dabei die x-Koordinate um die Zeichenbreite. Das Modusbyte wird unverändert weitergegeben (durchgeschleift). Beide Funktionen beziehen die Koordinatenwerte auf die linke obere Ecke der Ausgabe, die x-Koordinate wird auf die linke Bytegrenze abgerundet. Die Ausgabe erfolgt bei starken Vergrößerungen nicht gerade schnell, dieser Mangel wäre aber nur durch vollständige Programmierung in Maschinensprache zu beseitigen gewesen.

Durch Einführen eines zusätzlichen Parameters ist es möglich, die Textproportion, also das Verhältnis zwischen horizontaler und vertikaler Ausdehnung, für jede Ausgabe festzulegen, wodurch sich zum Beispiel Fettdruck sehr einfach realisieren ließe. Dazu müßte dieser Parameter der Variablen *'scan'* anstatt von *'size'* zugewiesen werden (oder auch:

'for (scan = 0; scan < size / 2; scan++)'



wodurch der Text zwar mit doppelter Breite, aber nur mit einfacher Höhe ausgegeben würde).

Ein Vorteil dieser Funktionen ist, daß sie auch unter TOS verwendbar sind. Die Funktion *'fontad()'* läßt sich bei den glücklichen Besitzern einer RAM-TOS-Version auch dazu verwenden, durch einfaches Einkopieren in das Betriebssystem der gesamte System-Font durch einen DEGAS-Font zu ersetzen. (Ich bin gerne dazu bereit, Interessierten einen einfachen Font-Editor sowie sehr brauchbare Fonts und das "Einkopierprogramm" für den 'AUTO-Ordner' zur Verfügung zu stellen.)

Für die Umsetzung in Programmiersprachen, die keine Möglichkeit des Verwendung von Line-A-Befehlen besitzen, sei angemerkt, daß die benötigten Fontadressen auch an folgenden Systemadressen zu finden sind (ST-Computer 2/88, Seite 42, Rainer KRACHT):

Pointer für Font-Daten der 8x16-Font: bei 0x607e 8x8-Font:
bei 0x41a8
Damit ergibt sich folgende Programmiermöglichkeit (Routine *'bchar()'*:

Pointerinitialisierung:

/ DieInitialisierung von 'fnt' fällt weg;*
für 'font':
/if (mode&64) font =*(char) 0x41a8+c;*
*else font= *(char**) 0x607e+c;*

Schleifenbedingung:

/ statt: *(int*)(fnt+82) */*
/ für 8x16:16 */*
/ für 8x8: 8 */*

```
 1:  extern long gemdos(), bios(), xbios();
 2:
 3:  #define   Tgettime()         (int)gemdos(0x2c)
 4:  #define   Cursconf(a,b)      xbios(21,a,b)
 5:  #define   Setcolor(a,b)      xbios(7,a,b)
 6:  #define   Kbshift(a)         bios(11,a)
 7:
 8:  #define   INVERS     128
 9:  #define   FNT2        64
10:  #define   ODER        32
11:
12:  long fontad(index)
13:  int index;
14:  {    register long *adr;
15:       asm {    move.l     A2,-(A7)
16:                dc.w       0xa000
17:                movea.l    A1, adr
18:                move.l     (A7)+,A2
19:           }
20:       return *(adr + index);
21:  }
22:
23:
24:
25:  int bchar(c, x_ko, y_ko, mode)
26:  int c, x_ko, y_ko, mode;
27:  {    int y, inv = mode & 128;
28:       register int x, size =mode&31, scan, op
                                         =mode&32;
29:       register char *screen, *font, *control;
30:       char work, transfer = 0, *fnt;
31:       extern long xbios();
32:       long fontad();
33:       screen =(char *)(xbios(3)+y_ko*80+(x_ko >> 3));
34:       control = (char *)(xbios(3) + 32000 - size *
                                              80);
35:       fnt = (char *)fontad(2 - ((mode >> 6) & 1));
36:       font = *(char **)(fnt + 76) + c;
37:       for (y = 0; y < *(int *)(fnt + 82); y++)
38:       {    work = inv ? ~*font : *font;
39:            for (x = 0; x < size * 8; x++)
40:            {    if ((work >> (7 - (x / size))) & 1)
                         transfer |= (128 >> (x % 8));
41:                 if (x % 8 == 7 && screen < control)
42:                 {  for (scan = 0; scan < size; scan++)
43:                      if (op) *(screen+scan*80) |
                                            =transfer;
44:                      else    *(screen + scan * 80)
                                            =transfer;
45:                      transfer = 0;
46:                      screen++;
47:                 }
48:            }
49:            font += 256;
50:            screen += (size * 79);
51:       }
52:       return size * 8;
53:  }
54:
55:
56:
57:  bstring(string, x, y, mode)
58:  char *string;
59:  int x, y, mode;
60:  {    while (*string) x += bchar(*string++, x, y,
                                              mode);
61:  }
62:
63:
64:
65:  main()
66:  {    char zeile[6];
67:       int zeit, i;
68:       Cursconf(0, 71);
69:       atari();
70:       for (i = 0; i < 16; i++)
71:            Setcolor(i,(unsigned)(~Setcolor(i,-1) &0x
                                              777));
72:       bstring("Beenden durch [Control]!", 224, 391,
                                          1| FNT2);
73:       while (!Kbshift(-1))
74:       {    zeit = Tgettime();
75:            sprintf(zeile, "%02d:%02d",
76:                               (zeit >> 11) &0x1f,
77:                               (zeit >>  5) 0x3f);
78:            bstring(zeile, 0, 70, 16);
79:       }
80:       for (i = 0; i < 16; i++)
81:            Setcolor(i,(unsigned)(~Setcolor(i,-1)& 0x
                                              777));
82:       Cursconf(1, 71);
83:  }
84:
85:
86:
87:  atari()
88:  {    int i;
89:       long wait;
90:       for (i = 1; i < 11; i++)
91:       {  bstring("\16\17 Atari", 320 - i * 32, 200 -
                                      i * 8,i|INVERS);
92:            for (wait = 500000; wait-;);
93:       }
94:       bstring("          ", 0, 120, 10);
95:  }
96:
97:
98:
99:
```

Listing zum Vergrößern des Zeichenmsatzes

Bildhaft aus- und eingepackt

In den vergangenen Monaten haben wir uns mit den unterschiedlichsten Bildformaten beschäftigt und diese näher beschrieben. In der vierten und damit letzten Folge unserer Bildformate haben wir uns dem IMAGIC-Format verschrieben. Dabei haben wir uns auf vielfachen Wunsch doch dazu entschlossen, nicht nur den Entpacker sondern auch den Packer für dieses Format zu veröffentlichen. Natürlich arbeitet der hier veröffentlichte Packer nicht mit allen Finessen wie in IMAGIC selbst, aber er packt so schon faszinierend gut. Aber zunächst wollen wir noch ein wenig der Theorie fröhnen...



Durchschaut

In den letzten Folgen unserer Reihe haben wir schon mehrfach festgestellt, daß ein Bild uns leider nicht den Gefallen tut, nur ganz bestimmte Bytes zu benutzen, sondern in den Bilddaten können immer alle Bytes enthalten sein, die es gibt;

von $00 für 8 weiße Punkte
über $55 oder $AA für grau
bis zu $FF = schwarz

Bitweise im Klartext: $55 = %01010101
(beispielweise, monochrom).

Ein Bild packen bedeutet, daß man Ketten gleicher Bytes zu möglichst wenigen Kontrollbytes zusammenfaßt, aus denen der Entpacker erkennt, wie er die alte Bildinformation wieder erstellen kann. Da jedoch alle 256 möglichen Bytes in den Bilddaten vorkommen können, muß man eine Codiervorschrift ersinnen, die es trotzdem ermöglicht, die Kontrollinformation in den gepackten Bilddaten zu erkennen. Die einfachste Möglichkeit wäre es natürlich, das 257. Byte zu verwenden, um die Kontrollinformation zu markieren. Nur leider ist es bis jetzt trotz intensivster Forschungsarbeiten nicht gelungen, dieses Byte zu entdecken. So bleiben uns praktisch nur noch zwei weitere Möglichkeiten:

Der Packer und der Zähler

Die erste Möglichkeit haben wir bereits bei den IFF- und VDI-Formaten kennengelernt: Jeder zusammenhängenden Kette von Informationen wird ein *"Zählerbyte"* vorangestellt, das besagt, wieviel reine Datenbytes jetzt folgen. Gleichzeitig besagt hierbei meist ein negatives Byte, daß eine Kette von m GLEICHEN Bytes folgt. Folgerichtig gibt es auch ein postives Byte, das die Anzahl der unterschiedlichen Bytes kodiert. Der Vorteil dieser Methode liegt darin, daß unser Code immer transparent bleibt, Steuerbytes und Datenbytes sind genau voneinander getrennt,der Auspacker weiß anhand der Zählerbytes genau, wie lang eine Kette von reinen Datenbytes ist.

Da ein Byte in einem solchen Fall nur

Werte von 0 .. 127 für positive Zahlen und -1 ... -128 für negative Zahlen annehmen kann, liegt hier auch der Nachteil der Methode: Die maximale Länge einer Kette, egal ob gepackt oder ungepackt, kann nur 127 Bytes betragen. Danach muß wieder ein neues Zählerbyte folgen. Das kann durchaus dazu führen, daß ein gepackter einmal auch länger als ein ungepackter Code wird. Die Unterbrechungen verlangsamen außerdem den Vorgang des Entpackens ein wenig. Ein weiterer Nachteil: Diese Methode läßt keine wirklichen "Codebäume" zu, mit denen man noch weitere Informationen verschlüsseln kann, wie man das zum Beispiel bei "direkter" Differenzkomprimierung benötigt.

Dem Code entflohen das ESCAPE-Byte

So ganz unsinnig, wie Sie meinen, ist die Idee vom 257. Byte gar nicht: Wenn es natürlich auch mit 8 Bits genau 256 (2 hoch 8) verschiedene Kombinationen gibt, so "bestimmt" man ganz einfach eines dieser Bytes zum *ESCAPE-Byte* (Fluchtbyte) - so etwas ähnliches hatten wir schon beim STAD-Format. Immer wenn in den komprimierten Daten dieses ESCAPE-Byte auftritt, erkennt der Entpacker daran, daß jetzt neue Steuerinformationen folgen. "Halt!" werden Sie sagen, "was passiert, wenn aber genau dieses ESCAPE-Byte in den Bilddaten vorkommt? Wir wollen doch nicht auf dieses schöne Byte im Bild verzichten?". Natürlich nicht! In diesem Fall schreiben wir das ESCAPE-Byte einfach ZWEIMAL hintereinander hin. Daran erkennt jetzt unser Entpacker, daß wirklich das ESCAPE-Byte gemeint ist und schreibt es nur EINMAL in das ausgepackte Zielbild.

So, somit ist das Unmögliche möglich gemacht: Wir haben '257 unterschiedliche Bytes' zur Verfügung! Wenn man jetzt ganz listig vorgeht, so bestimmt man dasjenige Byte zum ESCAPE-Byte, das am wenigsten häufig im ungepackten Bild vorkommt. Denn jedesmal, wenn das ESCAPE-Byte erscheint, bekommt unser Bilderpacker einen Schluckauf, weil dieses Byte ja verdoppelt werden muß. Da ist es natürlich sinnvoll, ein Byte auszuwählen, das diese Verdopplung möglichst selten erforderlich macht.

Wie man das ESCAPE-Byte bestimmt, können Sie beispielsweise schön in der Routine *GET_ESCAPE* sehen, die im Assemblercode des IMAGIC-Packers enthalten ist.

O Code-Baum, o Code-Baum ...

... du grünst nicht nur zur Winterszeit, nein auch im Sommer, wenn es schneit! Also was ist ein Codebaum? Ganz einfach: Eine Verzweigungs-Vorschrift. Unser ESCAPE-Byte sei die *"WURZEL"* unseres Baums. Dann bestimmt das darauffolgende Byte, auf welchem Ast wir uns weiter bewegen. Da ein Byte 256 Werte annehmen kann, gibt es also bereits 256 Äste und damit 256 Möglichkeiten zur Verzweigung! Ein Ast ist bereits vergeben: Das ESCAPE-Byte selbst noch einmal, das dann zu einem Datenbyte im Zielbild wird.

IMAGIC komprimiert:

Komprimierte Bilddaten aus dem Zeichen- und Animationsprogramm DENISE aus dem IMAGIC-Paket.

32 Bytes Farbpalette nach dem ATARI ST Standard,
Bytes 7 .. 38,

Auflösung LOW (*.IC1), MED (*.IC2), HIGH (*.IC3),

Dateilänge variabel, abhängig vom Bildinhalt,

Kennung:Bytes 1..4 = 'IMDC'

Bytes 5, 6 = Auflösung (0=LOW, 1=MED, 2=HIGH).

Da unser erzeugter Code möglichst kurz sein soll, ist es sinnvoll, die notwendigen Entscheidungen möglichst schnell zu treffen, also ein ganzes Bündel von Ästen zu einer Richtung zusammenzufassen: Nehmen wir an, eine Codeentscheidung bestehe immer aus drei Bytes:

<ESCAPE-Byte> <Zähler-Byte> <Datenbyte>

so sieht man sofort, daß es nicht sinnvoll ist, Wiederholungsketten von weniger als 4 Bytes Länge gepackt zu speichern, da sonst das Ergebnis länger als oder nur gleich lang sein würde wie die ungepackte Speicherung. Wenn wir den Wertebereich des Zähler-Bytes um Eins nach oben verschieben, können wir mit den Bytes ($03) .. ($FF) Zählerwerte von 4 .. 256 erreichen. Somit stehen die Zähler-Bytes ($01) und ($02) frei zu unserer Verfügung, da sie niemals auftreten können. Wir setzen Sie in unserem Codebaum ein:

Identische Bytes in einer Folge

Ein Byte *$nn* ($nn = $03 .. $FF) bedeutet, daß <nn+1> identische Bytes folgen: Der Auspacker nimmt das nächste Byte und packt es <nn+1> mal in das Zielbild aus. Ein Zählerbyte = ($01) bedeutet, daß der gesamte Zählwert um 256 Bytes erhöht wird, weitere ($01)-Bytes können folgen, womit wir den meist vorhandenen Nachteil, daß nur Wiederholungen kleiner gleich 256 möglich sind, umgangen haben. Nach ($00) folgt dann noch ein normales Zählerbyte, für den nicht durch 256 teilbaren Rest (für Kenner: Modulo 256) ...Schauen wir uns das an einem Beispiel an, wird es bestimmt verständlich:

($ESC)($01)($01)($00)($20)($55)
bedeutet demnach

"Es sind 256 + 256 + 33 = 545 Bytes mit dem Wert $55 auszupacken".

Ein Bild wird abhängig

Ein Zählerbyte ($02) hat eine weitere Sonderstellung: Es gilt als zweites ESCAPE-Byte, zur besseren Unterscheidung *SAME-Byte* genannt (Für alle, die der englischen Sprache nicht so gewachsen sind, sei am Rande erwähnt, daß SAME soviel wie 'ähnlich' bedeutet.). Tritt es auf, führt der Codebaum eine weitere Verzweigung durch: Es wird entschieden, ob folgende Teile des Bildes differenzkomprimiert sind (siehe unten), oder ob das Ende der komprimierten Daten erreicht ist. Es gelten jetzt die gleichen Regeln wie für das Zählerbyte für identische Bytefolgen: ($nn = $03 .. $FF)

bedeutet: <nn+1> folgende Bytes sind gleich wie im Basisbild, der Auspacker tut hier nichts anderes mehr, als brav die Adresse im Zielbild hochzuzählen. Ein Zählerbyte = ($01) bedeutet auch hier: Zählerwert um 256 Bytes erhöhen. Und auch an dieser Stelle wollen wir unseren interessierten Leser nicht ohne Beispiel lassen:

($ESC)($02)($01)($01)($00)($20) =
545 Bytes differenz komprimiert.

Die besondere Codekette ($ESC)-($02)($00) ist die Endemarke für unseren Auspacker, der sich daraufhin erschöpft zurücklehnt: Fertig!

Der kleine Unterschied Differenzpacken

Stellen Sie sich vor, Sie sind ein begeisterter Anhänger von JOHN WAYNE (oder einem nicht unbekannten Präsidenten) und möchten eine kleine Szene aus seinem bewegten Filmleben einmal auf Ihrem ATARI-Rechner nachempfinden: Sie entwerfen also folgendes Szenario: Die Wüste von Arizona, strahlendblauer Himmel, im Hintergrund ein Bergmassiv, ansonsten rötlicher Staub soweit das Auge reicht. John, als Meldereiter der Armee, reitet in gestrecktem Galopp von rechts kommend in das Bild. Ungefähr in der Bildmitte erblickt sein treues Pferd Rosalie eine Klapperschlange vor sich auf dem Boden, scheut auf - und John fliegt in hohem Bogen in den Sand. Es geht hier jetzt nur um das Pakken, nicht um das Niveau unserer Beispiele ...)

Nehmen wir also weiterhin an, die ganze Sequenz dauert so ca. 5 Sekunden, Sie benötigen also ungefähr 75 Einzelbilder, um sie halbwegs ruckfrei vorzuführen. Wenn Ihr ATARI ST jetzt nicht gerade mit 4 Megabyte Speicher ausgerüstet ist, wird es ziemlich eng im Speicher zugehen. Weil Ihr Bild so einen schönen blauen Himmel hat und sich der Sand in gleichmäßigem Rot präsentiert, können Sie beim Packen der Einzelbilder einen Gewinn von 50% erzielen. Macht 75 * 16000 Bytes = 1,2 Megabytes - immer noch zuviel für einen 1040 ST. Doch eigentlich sind sich alle Bilder Ihres kleinen Films doch sehr ähnlich: Der gesamte Bildhintergrund ist immer gleich, das bräuchte man doch nur einmal abzuspeichern ...sprachs und hatte auch schon eine Lösung parat, denn genau hier wird jetzt das Differenzpacken interessant! Es wird nur noch abgespeichert, was sich in einem Bild in Bezug auf ein anderes Bild ändert. Alles, was in beiden Bildern identisch ist, wird wie eine "transparente" Fläche (oder auch wie im Trickfilm eine Folie) beschrieben und kann dann wie bisher komprimiert werden. Jetzt erreichen Sie einen Gewinn zwischen 80% und 90% - die Animation paßt spielend in den Speicher ...

Bei der Differenzkomprimierung gibt es zwei unterschiedliche Techniken, deren Vorgehensweise wie folgt ist:

Exklusive Veroderung

1) Begonnen wird mit dem ersten Bild der Animation: Es wird ganz normal gepackt, wie bisher.

2) Das zweite Bild wird jetzt mit dem ersten Bild *"exklusiv verodert" (XOR)*, das heißt, es bleiben nur die Punkte stehen, die in beiden Bildern UNTERSCHIEDLICH sind.

3) Das Ergebnis wird dann gepackt.

4) Das dritte Bild wird mit dem zweiten Bild "exklusiv verodert" und so weiter.

Auf diese Weise erhalten wir eine gepackte Bildfolge, die nur aus den Unterschieden von aufeinanderfolgenden Bildern besteht. Beim Auspacken wird jedes Bild wieder auf seinen Vorgänger "exklusiv verodert", die ursprüngliche Bildfolge entsteht ...

Direktes Differenzpacken

Das zweite Verfahren ist die "direkte Differenzkomprimierung", wie sie vom IMAGIC-COMPILER beim Zusammensetzen eines Films eingesetzt wird:

1) Als erstes ist der Benutzer aufgefordert, ein BASISBILD zu bestimmen, das als Hintergrund der Animation gilt und sich möglichst wenig von allen weiteren Bildern unterscheidet: in unserem Beispiel also das Bild mit dem leeren Hintergrund, ohne die Akteure (John Wayne and his horse, you know).

2) Das Basisbild wird gepackt, wie bisher.

Der Header einer Imagic-Datei:

Bytes 1.. 4: 'IMDC' als ASCII-Text für "Imagic Delta Compressed"

Bytes 5.. 6: Auflösung 0=Low, 1=Medium, 2=High

Bytes 7..38: Farbpalette 16 Worte

Bytes 39..40: Datum der Erstellung im GEMDOS-Format (Tgetdate). 1)

Bytes 41..42: Uhrzeit der Erstellung im GEMDOS-Format (Tgettime).

Bytes 43..50: Name des Basisbilds. == 0 2)

Bytes 51..52: Länge der komprimierten Packdaten.

Bytes 53..56: Registriernummer. 3)

Bytes 57..64: reserviert. == 0 4)

1) Dadurch wird es möglich, immer festzustellen, wann ein Bild erstellt wurde. Datum und Uhrzeit einer Datei werden von GEMDOS nämlich leider bei der Erstellung einer Dateikopie immer neu gesetzt. Die Angabe ist optional, siedarf auch = 0 sein.

2) Derzeit wird die Differenzkomprimierung bei einzeln gespeicherten Bildern von IMAGIC nicht unterstützt. Der Platz ist für einespätere Erweiterung reserviert. Zur Zeit sind alle 8 Bytes = 0 zu setzen.

3) Es besteht die Möglichkeit, die Registriernummer des Programms einzutragen, mit dem das Bild erzeugt wurde. Damit lassen sich Raubkopien registrierter Programme besser zurückverfolgen. Die Angabe ist optional und kann auch = 0 sein.

4) Reserviert für spätere Erweiterungen. Sollte = 0 sein.

3) Bei jedem weiteren Folgebild wird jetzt in jedem kleinen Detail bestimmt,ob es gleich zu dem bekannten Basisbild ist. Dieser Arbeitschritt wird vom Packer direkt durchgeführt, die Bilder werden also nicht vorher verodert.

4) Gleiche Bildteile werden wie eine *"transparente"* Fläche in wenigen Bytes komprimiert.

5) Wenn erforderlich, kann das Basisbild gewechselt werden, es bleibt aber in der Regel für eine ganze Kette von Bildern das gleiche. Bedenken Sie aber, daß bei einer Veränderung des Basisbildes alle abhängigen Bilder neu komprimiert werden müssen.

Auf diese Weise erhalten wir eine Folge von Einzelbildern, die voneinander UNABHÄNGIG sind. Jedes Bild kann zusammen mit dem Basisbild für sich allein ausgepackt werden. Beim Auspakken wird immer erst eine Kopie des Basisbildes erstellt, in die der Auspacker die noch vorhandenen Änderungen schreibt. Dieses Verfahren hat einige Vorteile: Man kann einen Film jetzt nicht nur in fester Folge, sondern beliebig abspielen, vorwärts, rückwärts, ...! Weiterhin ist in den meisten Fällen das Ergebnis deutlich kürzer als bei der "XOR"-Komprimierung: Wenn sich das Pferd von einem Bild zum nächsten weiterbewegt, entstehen bei der Exklusiv-Veroderung Differenzdaten nicht nur an der Stelle, wo das Pferd neu gezeichnet, sondern auch dort, wo es aus dem letzten Bild herausgenommen werden muß. Nachteilig ist die direkte Differenzkomprimierung nur dann, wenn sich der gesamte Bildinhalt von Bild zu Bild geringfügig aber stetig ändert, der Unterschied zu dem festen Basisbild also immer größer wird. Hier wird es dann erforderlich sein, ab und zu ein neues Basisbild zu bestimmen.

Bildabtastung

Oh! Bei der Entwicklung von Strategien, Bilder möglichst gut zu packen, sind wir noch nicht am Ende! Bis jetzt läuft unser Packer schön brav von links nach rechts Zeile für Zeile das Quellbild ab und versucht dabei, möglichst lange Ketten von gleicher Information zu finden. Für das nächste Beispiel werden wir zum Praktiker und besorgen uns zunächst einen großes Blatt Papier, einen dicken Filzstift und eine Schere. Sie zeichnen viele verschiedene Linien, Muster etc. auf dieses Blatt, unter anderem auch mehrere große schwarze Rechtecke. Wenn Sie dieses Blatt Papier mit einer Schere in viele schmale horizontale Streifen zerschneiden und diese Streifen dann hintereinander zu einem langen Band auf den Boden legen, haben Sie eine Kette von Informationen so, wie unser Packer bis jetzt ein Bild Zeile für Zeile abtastet. Ihre schwarzen Rechtecke erscheinen auf dem Papierband als kurze schwarze Streifen, immer wieder unterbrochen von den unregelmäßigen Linien und Mustern, die sonst auf dem Papier zu sehen waren. Entsprechend kurz sind auch die Ketten zusammenhängender Information, die der Packer auf einmal verschlüsseln kann. Wenn wir jedoch ein Bild als zweidimensionale Datenstruktur auffassen und unseren Packer befähigen, das Bild in mehrere Rechtecke zu unterteilen, die er dann nacheinander abtastet, erhalten wir auf einmal viel öfters zusammenhängende Blöcke mit gleicher Information. Das ist genau die Arbeitsweise, mit der die verschiedenen Packer von IMAGIC ein Bild zweidimensional abtasten. Unter glücklichen Umständen kann die Komprimierungsrate dabei noch einmal um 50% gegenüber der linearen Abtastung gesteigert werden.

Da IMAGIC nicht weiß, welche Abtastung für ein Bild das optimale Ergebnis bringt, wurden aus den vielen hundert Möglichkeiten durch statistische Versuche die 19 effektivsten Algorithmen ausgewählt. Sie entsprechen den IMAGIC-Packern #2 .. #20. Packer #1 ist der lineare Packer, der ein Bild in horizontalen Streifen abtastet. IMAGIC probiert beim Pakken eines Bildes einfach alle 20 Algorithmen durch und merkt sich die Länge der entstehenden, komprimierten Daten. Der Algorithmus mit der geringsten Länge gewinnt.

IN MEDIAS RES

Der Aufbau einer IMAGIC-Bilddatei

Kommen wir zu den Fakten und schauen uns den Aufbau einer IMAGIC-Datei an. IMAGIC-Bilder haben einen 64 Byte-Header mit folgenden Informationen: Die ersten vier Bytes enthalten den Text *IMDC*, was eine Erkennung der Datei ermöglicht und ein Akronym für *"IMAGIC DELTA COMPRESSED"* darstellt. In den Bytes fünf und sechs finden wir die Auflösung des Bildes verschlüsselt, wobei 0 *low*, 1 *medium* und 2 *high* bedeutet. Die in der IMG-Datei so stiefmütterlich behandelte Farbpalette bringen wir in den Bytes 7 bis 38 unter, auf die in den beiden folgenden Bytes das Datum der Erstellung im GEMDOS-Format folgt. Dadurch wird es möglich, immer festzustellen, wann ein Bild erstellt wurde. Datum und Uhrzeit einer Datei werden von GEMDOS nämlich leider bei der Erstellung einer Dateikopie immer neu gesetzt. Die Angabe ist optional, sie darf auch = 0 sein. Für den Namen eines Bildes sehen wir acht Buchstaben vor, der also mit im Header vorhanden ist. Derzeit wird die Differenzkomprimierung bei einzeln gespeicherten Bildern von IMAGIC nicht unterstützt. Der Platz ist für eine spätere Erweiterung reserviert - zur Zeit sind alle 8 Bytes = 0 zu setzen. Die Länge der komprimierten Packdaten halten wir in Bytes 51 und 52 fest, während die Bytes 53 bis 56 eine Seriennummer enthalten. Es besteht die Möglichkeit, die Registriernummer des Programms einzutragen, mit dem das Bild erzeugt wurde. Damit lassen sich Raubkopien registrierter Programme besser zurückverfolgen; auch diese Angabe ist optional und kann = 0 sein. Die letzten vier Bytes sind für spätere Anwendungen reserviert und sollten daher Null sein.

Der IMAGIC-Packer!

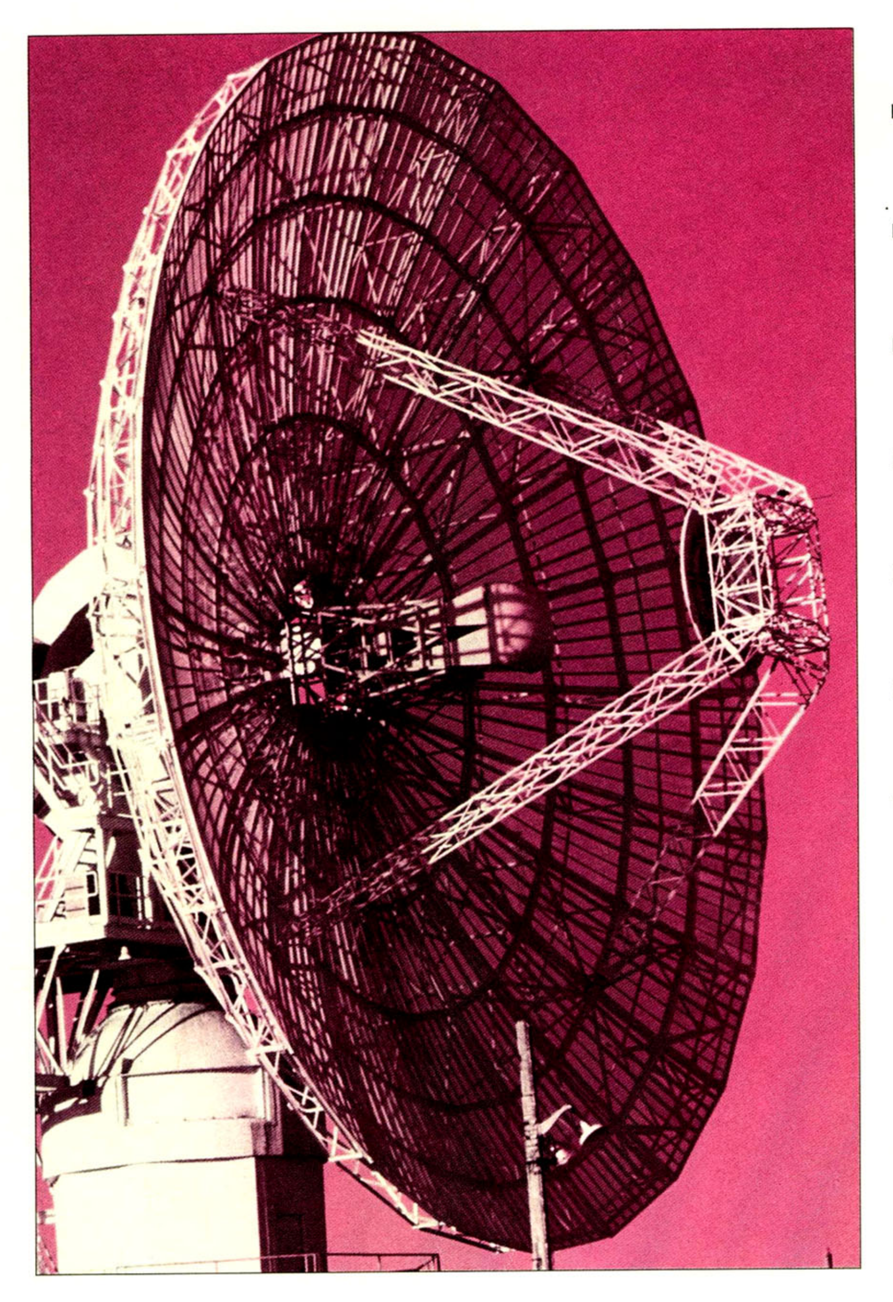
Wir haben einige Briefe bekommen, deren Absender sich wünschten, daß wir einen Packer veröffentlichen. Wir haben uns entschlossen, den Linearpacker von IMAGIC zu veröffentlichen (Jörg und Alex sei Dank). Dieser Packer erzeugt Bilder, die genau den Konventionen des IMAGIC-Formats entsprechen. Gepackt werden können Bilder aller Auflösungen(!) des ATARI ST. Der vorgestellte Algorithmus entspricht dabei dem Packer #1 aus dem IMAGIC-Paket und führt eine lineare Abtastung des Quellbildes durch. Eine Differenzkomprimierung wird für Einzelbilder nicht durchgeführt. Auch wenn wir ihn veröffentlichen, die Erklärung soll nicht über die Dokumentation hinausgehen - die Grundlagen zum Verständnis des Packers sind in den letzten Folgen der ST-Ecke gelegt worden.

Der IMAGIC-Auspacker

...packt alle mit IMAGIC oder dem vorgestellten Packer erstellten Bilder. Er ist auch bereits für das Auspacken differenzkomprimierter Bilder vorbereitet.

Resümee

Eine noch höhere Packdichte ist machbar, wenn man Datenstrukturen erkennt und packt, die kleiner als ein Byte sind (Bitpacker). Dabei kann dann ein noch optimalerer Codebaum erstellt werden, wenn



...UND PLÖTZLICH HAT IHR ATARI ST ODER MEGA ST EINE BEGEGNUNG DER BESONDEREN ART: VORTEX HD PLUS

Festplatten-Systeme von 20 bis 120 MB!

Das hat es bis jetzt noch nicht gegeben: Ein Festplatten-Programm für den ATARI ST bzw. MEGA ST mit Kapazitäten von 20 bis 120 MB formatierte Speicherkapazität!

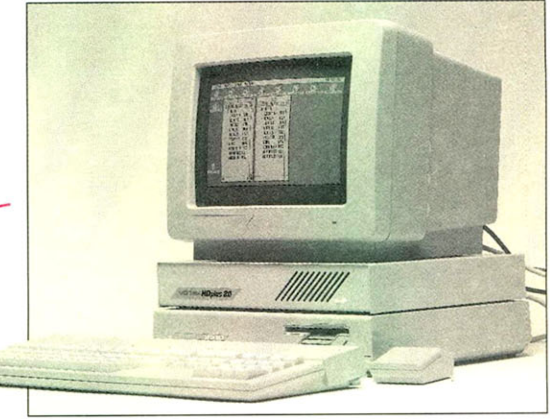
Weitere Vorteile

- Cache-Memory
- Auto-Parker
- bis zu 16 Partitionen
- Disketten-Backup-Programm
- bootfähig

Natürlich im ATARI-Design und in vortex-Qualität. Komplett anschlußfertig mit System-Diskette, Buskabel und deutschem Handbuch.

Holen Sie sich die kompletten Informationen. Sofort!





I·N·F·O·S·C·H·E·C·K

Senden Sie mir umgehend alle Informationen über Ihr HDplus-Programm und nennen Sie mir den nächstgelegenen vortex-Vertragshändler.



...UND PLÖTZLICH LEISTET IHR COMPUTER MEHR

vortex Computersysteme GmbH
Falterstraße 51–53 · 7101 Flein · Telefon (07131) 52061

beispielsweise die Häufigkeiten der vorkommenden Bitfolgen im Bild beachtet werden. Schön gelöst ist das zum Beispiel bei der *ARC* (Archive Utility), einem PD-Programm zur Komprimierung und Archivierung von Daten. Für den praktischen Einsatz bei IMAGIC ist die Anwendung der Bit-Komprimierung leider nicht geeignet, weil das Packen und Entpacken sehr viel Zeit erfordern.

The Final Curtain

Nun fällt der Vorhang über unseren bildlichen Streifzug, und wir hoffen, daß wir ein wenig Licht in das Dunkel gebracht haben. Einige unterschiedliche Formate haben wir nicht nur als Format selbst, sondern auch mit den entsprechenden Routinen veröffentlicht, so daß sie direkt verwendet werden können. Wichtig war es uns, die Vor- und Nachteile der Packformate aufzuzeigen, und, daß der Leser bemerkt, daß es nicht DEN Packalgorithmus gibt, sondern nur einen für das spezielle Problem ausgesuchten. Wie überall im Leben muß man Kompromisse schließen und sich (natürlich im übertragenen Sinne) das beste Preis-/Leistungsverhältnis heraussuchen. Viel Spaß und Erfolg dabei wünschen Jörg Drücker und Stefan Höhn, die sich vielleicht irgendwann mit einer BILD-Nachlese wieder zurückmelden.

Die ST-Ecke widmet sich in den nächsten Monaten wieder ausgesuchten Einzelthemen, und ich kann Ihnen ankündigen, daß wir ein paar nette Sachen auf Lager haben. Versprochen!

(Jörg Drücker/SH)

Literaturhinweise für Besessene:

Codierungsregeln nach Shannon, Huffman und Lempel-Zev in diversen Informatik-Büchern.

Dokumentation zur Archive-Utility. (PD, auf Diskette).

```
  1: * modul AUSPACK.S
  2:
  3:
  4:
  5: * Assemblermodul zum Auspacken von
  6: * verschiedenen gepackten Bildformaten.
  7:
  8: * Assembler: Metacomco MACRO Assembler
  9: * für andere Assembler den Stern am Anfang
 10: * durch Semikolon ersetzen
 11:
 12:
 13:
 14:
 15: * Originalauszug aus dem Grafikpaket IMAGIC
 16: * von APPLICATION SYSTEMS /// HEIDELBERG.
 17:
 18:
 19: * Version 1.0
 20:
 21: * verfasst  am  8- 8-1988  von Jörg Drücker
 22: * erweitert am 15- 9-1988  GEM VDI und STAD Formate.
 23: * erweitert am 22-10-1988  IMAGIC Format.
 24:
 25: * Copyright (c) 1988 by IMAGIC GRAFIK.
 26:
 27:
 28:
 29: *                                         *
 30: *        IMAGIC BILDER                    *
 31: *        auspacken                        *
 32: *                                         *
 33:
 34:
 35:        xdef    DEC_IMAG
 36:
 37: DEC_IMAG: bsr       GET_PAR    * Parameter holen
 38:
 39:           cmpi.l    #'IMDC',(a0)* check header
 40:           bne       ERR_DONE    * not IMAGIC
                                      compressed.
 41:
 42:           bsr       ERASE_PIC
 43:           lea       64(a0),a0   * skip header
 44:           bsr       IMAG_DECOMP
 45:           bra       ALL_DONE
 46:
 47:
 48:
 49: *                                         *
 50: *        Fehlerausgang                    *
 51: *        Booleanwert FALSE zurückgeben    *
 52: *                                         *
 53:
 54:
 55: ERR_DONE:       moveq     #0,d0        * "error"
 56:                 bra.s     RETURN
 57:
 58:
 59:
 60: *                                       *
 61: *       Normalausgang                   *
 62: *       Booleanwert TRUE  zurückgeben   *
 63: *                                       *
 64:
 65:
 66: ALL_DONE:  moveq     #1,d0         * "no error"
 67:
 68:
 69:
 70: RETURN:    movem.l   (sp)+,d7/a6   * restore d7/a6
 71:
 72:            movea.l   (sp)+,a0
 73:            lea       12(sp),sp     * cleanup stack
 74:            jmp       (a0)          * rts
 75:
 76:
 77:
 78: *                                       *
 79: *       Hilfsfunktion                   *
 80: *       Parameter vom Stack holen       *
 81: *       Register korrekt setzen         *
 82: *                                       *
 83:
 84:
 85: GET_PAR: lea       8(sp),a4      * Zeiger auf
                                        Parameter
 86:
 87:          move.w    (a4)+,d0      * RESOLUTION
 
 88:          move.w    (a4)+,d6      * PICLEN
 89:          movea.l   (a4)+,a1      * PICTURE
 90:          movea.l   (a4)+,a0      * COMPRESSED DATA
 91:
 92:          move.l    (sp)+,a5      * Return Adresse
 93:          movem.l   d7/a6,-(sp)   * Register retten
 94:
 95:          jmp       (a5)
 96:
 97:
 98:
 99: *                                       *
100: *       Hilfsfunktion                   *
101: *       Bildinhalt löschen              *
102: *                                       *
103:
104:
105: * a1 = picture
106: * d6 = picturelen ( 32000 or more, but always a
         multiple of 64 bytes ! )
107:
108: ERASE_PIC: movem.l   d0-d7/a1-a2,-(sp)
109:
110:           moveq     #0,d0
```

Listing geht weiter...

```
111:            moveq     #0,d1
112:            moveq     #0,d2
113:            moveq     #0,d3
114:            moveq     #0,d4
115:            moveq     #0,d5
116:            moveq     #0,d7
117:            move.l    d0,a2            * "0" ins a2
118:
119:            adda.w    d6,a1            * upper border
120:            lsr.w     #6,d6            * picturelen
                                             div 64
121:            subq.w    #1,d6            * Zähler -1
122:
123:  clrpic:   movem.l   d0-d5/d7/a2,-(a1) * 32 bytes
124:            movem.l   d0-d5/d7/a2,-(a1) * 32 bytes
125:            dbra      d6,clrpic
126:
127:            movem.l   (sp)+,d0-d7/a1-a2
128:            rts
129:
130: *==============================================*
131: *                                              *
132: *           Routinen zur Dekomprimierung       *
133:
134: *                                              *
135: *           IMAGIC Dekomprimierung             *
136:
137: *                                              *
138: *           (c) 1987 by Jörg Drücker.          *
139:
140:
141: * a0 = IMAGIC komprimierter Code
142: * a1 = Zielbild
143: *
144: * d6 = Anzahl Bytes im Zielbild ( = 32000 )
     "piclen"
145:
146:
147:  IMAG_DECOMP:
148:
149:
150: * Teil I.
151: *
152: * Entscheiden, ob IMAGIC Bild gepackt ist:
153:
154:           tst.b     (a0)              * 1. Byte
155:           bne.s     DO_UNSQZ  * Null ? -> dann
                        Bilddaten kopieren !
156:
157:
158:
159: * Teil Ib.
160: *
161: * Nicht komprimierte Bilddaten einfach
162: * kopieren.
163: * Beachte: Die Bilddaten liegen nicht auf
164: * einer geraden Addresse, also ist ein
165: * Byte-Copy nötig !
166:
167:           addq.l    #1,a0
168:           lsr.w     #2,d6
169:           subq.w    #1,d6          * piclen/4 -1
170:
171:  COPY_L2: move.b    (a0)+,(a1)+   * BYTE COPY
                                         durchführen
172:           move.b    (a0)+,(a1)+
173:           move.b    (a0)+,(a1)+
174:           move.b    (a0)+,(a1)+
175:           dbra      d6,COPY_L2
176:
177:           rts          * Nach Hause telefonieren ...
178:
179:
180: * Teil II.
181: *
182: * Dekomprimierung vorbereiten:
183:
184:  DO_UNSQZ: link     a2,#-8   * lokalen BSS linken
185:
186:           move.l    a1,-4(a2) * obere Bilddaten
                                    grenze errechnen
187:           movea.l   a1,a3
188:           add.w     d6,a3 * untere Bilddatengrenze
                              errechnen
189:           move.l    a3,-8(a2)
190:
191:           moveq     #0,d1        * Register löschen
192:           moveq     #0,d2
193:
194:
195: * Teil III.
196: *
197: * Los geht's - codierte Daten auspacken:
198:
199:           move.b    (a0)+,d1
200:           move.b    (a0)+,d2
201:           move.b    (a0)+,d7
202:           mulu      #80,d2
203:           cmp.b     #-1,d1
204:           bne.s     NOT_NEGATIVE
205:
206:           move.w    d6,d1
207:           move.w    #1,d2
208:
209:  NOT_NEGATIVE:
210:           movea.w   d2,a4
211:           move.w    d2,d6
212:           subq.w    #1,d6
213:           move.w    d1,d5
214:           subq.w    #1,d5
215:           movea.w   d5,a3
216:           neg.w     d1
217:           muls      d2,d1
218:           addq.l    #1,d1
219:           movea.l   d1,a5
220:           muls      d5,d2
221:           movea.l   d2,a6
222:           moveq     #1,d1
223:           moveq     #3,d2
224:           moveq     #2,d4
225:           moveq     #0,d0
226:
227:  EX_LOOP: move.b    (a0)+,d0
228:           cmp.b     d0,d7             * (ESC) byte
229:           beq.s     ZERO_1
230:
231:  ZERO_2:  cmpa.l    -4(a2),a1
232:           bmi       IMAG_END        * Untergrenze
                                           überprüfen
233:           cmpa.l    -8(a2),a1
234:           bpl       IMAG_END        * Obergrenze
                                           überprüfen
235:
236:           move.b    d0,(a1)   * Ein Byte
                                   Zielbild schreiben
237:
238:           adda.l    a4,a1
239:           dbra      d5,EX_LOOP
240:           move.w    a3,d5
241:           adda.l    a5,a1
242:           dbra      d6,EX_LOOP
243:           move.w    a4,d6
244:           subq.w    #1,d6
245:           adda.l    a6,a1
246:           bra.s     EX_LOOP
247:
248:
249:
250:  ZERO_1:  move.b    (a0)+,d0 * Zähler oder
                         doppeltes (ESC) Byte ?
251:
252:           cmp.b     d0,d7
253:           beq.s     ZERO_2
254:
255:           moveq     #0,d3 * "multiple" Zähler
                                 löschen
256:           cmp.w     d2,d0
257:           bpl.s     ADD_BYTE
258:
259:
260:
261:  CHK_02:  cmp.b     d4,d0    * (SAM) - Markierung ?
262:           bne.s     CHK_01
263:
264:           move.b    (a0)+,d0   * Zähler holen
265:           beq.s     IMAG_END  * "(ESC)(SAM)00" -
                                     > Ende !
266:
267:
268:
269:           cmp.w     d2,d0
```

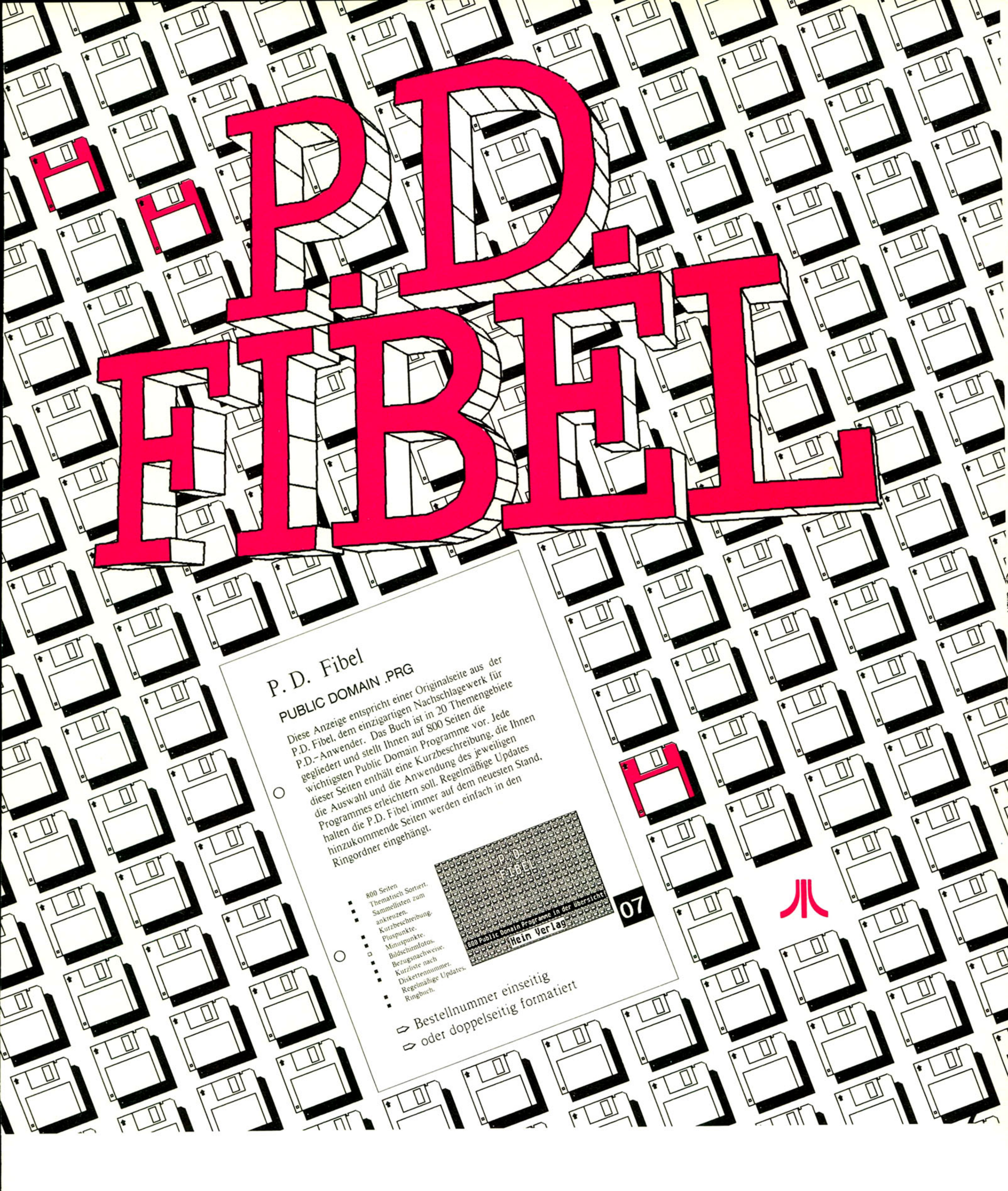
Listing geht weiter...

```
270:            bpl.s      ADD_S_BYTE
271:
272:            cmp.b      d4,d0
273:            bne.s      CHK_S_01
274:
275:  CHK_S_00: move.b     (a0)+,d0
276:            beq.s      EX_LOOP
277:            bra.s      CHK_S_00
278:
279:  CHK_S_01: cmp.b      d1,d0
280:            bne.s      SKIP_S_EOM
281:
282:            addi.w     #256,d3        * Zähler +256
283:
284:            move.b     (a0)+,d0
285:            bra.s      CHK_S_01 * nächstes Zählerbyte
286:
287:  SKIP_S_EOM:
288:            move.b     (a0)+,d0 * (EOM)-Byte weglesen
289:
290:  ADD_S_BYTE:
291:            add.w      d0,d3   * Zählerrest addieren
292:
293:
294:
295:  SAME_LOOP:
296:
297: * differenzkomprimiertes Zielbild:
298: * Die entsprech. Anzahl Bytes ist identisch wie im
299: * Basisbild.
300:
301:            adda.l     a4,a1
302:            dbra       d5,SAME_INCR
303:            move.w     a3,d5
304:            adda.l     a5,a1
305:            dbra       d6,SAME_INCR
306:            move.w     a4,d6
307:            subq.w     #1,d6
308:            adda.l     a6,a1
309:
310:  SAME_INCR:
311:            dbra       d3,SAME_LOOP
312:            bra.s      EX_LOOP
313:
314:
315:
316:  CHK_01:   cmp.b      d1,d0
317:            bne.s      SKIP_EOM
318:
319:            addi.w     #256,d3      * Zähler +256
320:            move.b     (a0)+,d0
321:            bra.s      CHK_01   * nächstes Zählerbyte
322:
323:  SKIP_EOM: move.b     (a0)+,d0 * (EOM)-Byte weglesen
324:
325:  ADD_BYTE: add.w      d0,d3   * Zählerrest addieren
326:
327:            move.b     (a0)+,d0  * Datenbyte
328:
329:
330:
331:  UNSQ_LOOP:
332:
333:  * Folgeketten-Komprimierung:
334:  * Die entsprechende Anzahl Bytes ist identisch,
335:  * daher wird das Datenbyte n-mal in das Zielbild
336:  * geschrieben.
337:
338:            cmpa.l     -4(a2),a1
339:            bmi.s      IMAG_END       * untere Grenze
                                            prüfen
340:            cmpa.l     -8(a2),a1
341:            bpl.s      IMAG_END       * obere Grenze
                                            prüfen
342:
343:            move.b     d0,(a1)        * Zielbyte
                                            schreiben
344:
345:            adda.l     a4,a1
346:            dbra       d5,UNSQ_INCR
347:            move.w     a3,d5
348:            adda.l     a5,a1
349:            dbra       d6,UNSQ_INCR
350:            move.w     a4,d6
351:            subq.w     #1,d6
352:            adda.l     a6,a1
353:
354:  UNSQ_INCR:
355:            dbra       d3,UNSQ_LOOP
356:            bra        EX_LOOP
357:
358:
359:  *
360:  * Auspackvorgang beendet:
361:
362:  IMAG_END: unlk       a2  * lokalen BSS freigeben
363:            rts          * home, sweet home again
364:
365:  *=========================================*
366:
367:            end
368:
```

Listing 1: Das Auspackprogramm

```
 1:
 2:  programm BILDEINLESEN ( input, output )
 3:
 4:  { Diese Programmteile gehören zu einen PASCAL-
     Programm, das in vorangegangenen Folgen der
 5:   ST-Ecke schon auszugsweise veröffentlicht
 6:   wurde. Diese Zeile sind allein natürlich
 7:   nicht lauffähig ! }
 8:
 9:  {...in den Copyright-Block }
10:
11:    erweitert am 22-20-1988  IMAGIC Einpacker.
12:
13:  {...bei den globalen Variablen }
14:
15:         DATEILAENGE : integer;
16:         ANTWORT     : char;
17:  {...hinter COL_VDI }
18:
19:  { Assemblerroutine zur Codierung
20:    in das gepackte IMAGIC-Format:
21:  }
22:
23:  function SQZ_IMAG (PICTURE, COMP : DATA_POINTER) :
     integer;
24:           external;
25:
26:  {...hinter HOLE_FARBEN }
27:
28:  function SETZE_BILDDATEN ( PICTURE, WRITEBUF :
        DATA_POINTER;
29:                             FARBEN   : COLOUR_PTR )
                                         : integer;
30:
31:  { Erstelle Bild im IMAGIC - Format: }
32:
33:  type IMAGIC_HEADER = packed record
34:
35:       { 64 Bytes IMAGIC compressed file header: }
36:
37:          ID      : packed array [ 1..4 ] of char;
38:
39:          ES      : integer;
40:          OLOR    : COLOUR_DATA;
41:
42:          ATE     : integer;
43:          IME     : integer;
44:
45:          ASE     : packed array [ 1..8 ] of char;
46:
47:          ENGTH   : integer;
48:          EGIS    : long_integer;
49:
50:          es_1    : long_integer;
```

Listing geht weiter...



Preise sind unverbindlich empfohlene Verkaufspreise.

Heim Verlag

Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

BESTELLCOUPON einsenden an Heim Verlag · Heidelberger Landstr. 194 6100 Darmstadt-Eberstadt

Bitte senden Sie mir ____ Stück **PD-FIBEL** für nur **DM 59,-** per Stück zzgl. DM 5,- Versandkosten (unabhängig von der Bestellmenge). Zahlung: ☐ Nachnahme ☐ Scheck liegt bei ☐ per Vorauszahlung

Name ____________ Vorname ____________ PLZ, Ort ____________

Str., Hausn. ____________ Unterschrift ____________

Benutzen Sie auch die in ST-COMPUTER vorhandene Bestellkarte.

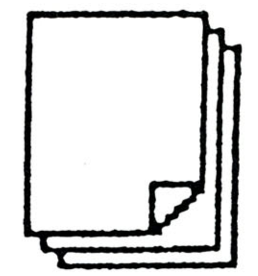
```
51:          es_2  : long_integer    {-reserved- }
52:
53:  end;
54:
55:  var IMAGIC : packed record case boolean of
56:
57:         false : ( HEADER : ^IMAGIC_HEADER);
58:         true  : ( DATA   : DATA_POINTER )
59:
60:  end;
61:
62:       SQZLEN,
63:       I       : integer;
64:
65:
66:  { Zusätzliche Systemfunktionen:
67:  }
68:
69:  function Tgetdate : integer;
70:
71:  { Systemdatum holen }
72:
73:          gemdos ( $2A );
74:
75:  function Tgettime : integer;
76:
77:  { Systemzeit holen }
78:
79:          gemdos ( $2C );
80:
81:
82:  begin
83:
84:       SQZLEN := SQZ_IMAG ( PICTURE,
85:                            ADDR_OFFSET ( WRITEBUF,
                               64) );  { Bild packen }
86:
87:       IMAGIC.DATA := WRITEBUF;
88:
89:       with IMAGIC.HEADER^ do begin
90:
91:            { IMAGIC - Dateiheader erstellen: }
92:
93:            ID      := 'IMDC';      { "IMAGIC DELTA
                                          COMPRESSED" }
94:
95:            RES     := AUFLOESUNG; { Bildsschirm  }
96:            COLOR   := FARBEN^;    { Farbpalette  }
97:
98:            DATE    := Tgetdate; {Datum der Erstel.}
99:            TIME    := Tgettime;{Uhrzeit der Erstel.}
100:
101:           for I := 1 to 8 do  { kein Basisbild  }
102:
103:                BASE [ I ] := chr ( 0 );
104:
105:           LENGTH := SQZLEN; { Länge der
                                  komprimierten Daten }
106:
107:           REGIS  := $4711;  { Registriernummer
                                   unseres Programms}
108:
109:           res_1  := 0;      { reserviert für
                                   Erweiterungen     }
110:           res_2  := 0
111:
112:      end;
113:
114:      SETZE_BILDDATEN := SQZLEN + 64 { gesamte
                                           Datenlänge  }
115:
116:  end; { SETZE_BILDDATEN }
117:
118:  procedure SPEICHERE_BILD (BILDBUF : DATA_POINTER;
119:                            LAENGE  : long_integer);
120:
121:  { Speichere gepacktes IMAGIC-Bild: }
122:
123:  var OK     : boolean;
124:
125:      POINT,
126:      SLASH,
127:      I      : integer;
128:
129:      ZIELDATEI : packed file of byte;
130:
131:  { Zusätzliche Systemfunktionen:
132:  }
133:
134:  function fwrite ( handle : integer; count :
                                        long_integer;
135:                    buffer : DATA_POINTER ) :
                                        long_integer;
136:
137:  { Schreiben von Daten in eine Datei }
138:
139:          gemdos ( $40 );
140:
141:
142:  begin
143:
144:       POINT := 0;
145:       SLASH := 0;
146:
147: { Suche LETZTEN Punkt "." und Backslash "\" im
       Dateinamen: }
148:
149:       for I := length ( DATEINAME ) downto 1 do
                  begin
150:
151:            if ( DATEINAME [ I ] = '.' ) and
                   ( POINT = 0 ) then
152:
153:                 POINT := I;
154:
155:            if ( DATEINAME [ I ] = '\' ) and
                   ( SLASH = 0 ) then
156:
157:                 SLASH := I
158:
159:       end;
160:
161:       if POINT > SLASH then
162:
163:     { Dateiname hat eine Extension,
164:       sonst ist die Extension im Ordnernamen }
165:
166:            { lösche vorhandene Extension: }
167:
168:            delete ( DATEINAME, POINT, length
                        ( DATEINAME ) - POINT +1 );
169:
170:    { setze korrekte Extension für IMAGIC - Bild:
171:
172:
173:            *.IC1 - für niedere Auflösung,
174:            *.IC2 - für mittlere Auflösung,
175:            *.IC3 - für hohe Auflösung
176:
177:       }
178:
179:       DATEINAME := concat ( DATEINAME, '.IC',
180:                  chr ( ord ( '1' ) + AUFLOESUNG )
);
181:
182:       { Zieldatei erstellen, ggf. vorher löschen:
183:       }
184:
185:       rewrite ( ZIELDATEI, DATEINAME );
186:
187:       OK := io_result = 0;
188:
189:       if OK then
190:
191:            { Dateiinhalt schreiben:
192:            }
193:
194:            OK := fwrite ( handle ( ZIELDATEI ),
                     LAENGE,BILDBUF ) = LAENGE;
195:
196:       if not OK then begin
197:
198:            { Im Fehlerfall unbedingt versuchen,
199:              Zieldatei wieder zu löschen:
200:            }
201:
202:            erase ( ZIELDATEI );
203:
204:            writeln ( chr ( 27 ), 'E', chr ( 7 ) );
205:                     { Bildschirm löschen, Warnton }
```

```
206:
207:          writeln ( 'Fehler beim Schreiben der
                  Bilddatei !' );
208:          writeln;
209:          writeln ( 'Weiter mit <RETURN>' );
210:          readln
211:
212:       end
213:
214:  end; { SPEICHERE_BILD }
215:
216:  {... im Hauptprogramm  hinter HOLE_FARBEN,
        setpalette,HOLE_BILDDATEN }
217:
218:      { Bildinhalt im IMAGIC Format packen:
219:      }
220:
221:      DATEILAENGE := SETZE_BILDDATEN ( BILDSCHIRM,
222:                                      LESEBUFFER.
                                                DATA,
223:                                      BILDFARBE);
224:
225:
226:      readln; { warte nur auf Tastendruck }
227:
228:
229:  {...hinter write(chr(27),'e'); }
230:
231:      if PIC_TYPE <> P_IMAGIC then begin
232:
233:     { Cursor in die unterste Zeile, Zeile löschen
234:
235:          }
236:
237:          write ( chr ( 27 ), 'Y', chr ( 32+24 ),
238:                  chr ( 32 ),chr ( 27 ), 'K' );
239:
240:          write   ( 'Speichern im IMAGIC
                       Format ? (J/N): ' );
241:          readln  ( ANTWORT );
242:
243:          setpalette ( SYSTEMFARBE );
244:          { Ursprüngliche Palette wiederherstellen }
245:
246:          if ( ANTWORT = 'J' ) or ( ANTWORT ='j' )
              then
247:
248:       { Bild als gepacktes IMAGIC-Bild abspeichern
249:       }
250:
251:             SPEICHERE_BILD ( LESEBUFFER.DATA
                     , DATEILAENGE )
252:
253:       end
254:       else
255:
256:          setpalette ( SYSTEMFARBE );
257:         { Ursprüngliche Palette wiederherstellen }
258:
259:{Das war der letzte Teil unseres PASCAL-Programms }
```

Listing 2: Das Bildeinleseprogramm

```
 1:  * modul EINPACK.S
 2:
 3:
 4:  *
 5:  * Assemblermodul IMAGIC Packer.
 6:
 7:  * Assembler: Metacomco MACRO Assembler
 8:  * für andere Assembler den Stern am Anfang
 9:  * durch Semikolon ersetzen
10:
11:  *
12:
13:
14:
15:  * Originalauszug aus dem Grafikpaket IMAGIC
16:  * von APPLICATION SYSTEMS /// HEIDELBERG.
17:
18:
19:  * Version 1.0
20:
21:  * verfasst  am 22-10-1988  von Jörg Drücker
22:
23:  * Copyright (c) 1988 by IMAGIC GRAFIK.
24:
25:
26:
27: PICTURELEN:    equ     32000      * Bildgröße
28:
30:
31:
32:  *
33:  *                IMAGIC Komprimierung
34:
35:  *
36:  *                (c) 1987 by Jörg Drücker.
37:  *
38:
39:           xdef      SQZ_IMAG
40:
41: END_SQZ:        equ      31     ( End-of-data Flag )
42:
43: PICTURE:        equ      8
44: DESTINATION:    equ      4
45:
46: *================================================
47:
48: SQZ_IMAG:
49:
50:
51:  * Teil I.
52:  *
53:  * (ESC)-Byte ermitteln:
54:
55:          movea.l   PICTURE+8(sp),a0 * Bilddaten
56:          bsr       GET_ESCAPE
                       * ermittle (ESC)-Byte -> d0.b
57:
58:          moveq     #0,d5
59:          move.b    d0,d5              * d5 = (ESC)
60:
61:
62:  * Teil II.
63:  *
64:  * Datenaddressen ermitteln,
65:  * Komprimierung vorbereiten:
66:
67:          movea.l   PICTURE(sp),a0  * Bilddaten
68:          movea.l   DESTINATION(sp),a1
                               * kompr. Zielgebiet
69:
70:          move.w    #PICTURELEN-1,d6 * Bytezähler
71:          move.w    #256,d4
72:          moveq     #3,d2
73:
74:
75:  * Teil III.
76:  *
77:  * Quellbild im IMAGIC-Format codieren:
78:
79:          move.b    (a0),d0         * Byte holen
80:          moveq     #-1,d3  * Initialwert Zähler
81:          move.b    d3,(a1)+
82:          move.b    #1,(a1)+
83:          move.b    d5,(a1)+
84:          bra.s     GET_NEXT * nächstes Byte holen
85:
86:
87:
88: RECOUNT:  tst.l     d5 * Ende der Daten erreicht ?
89:          bmi       END_SQUEEZE
90:
91:          moveq     #0,d3 * Zähler rücksetzen
92:
93:
```

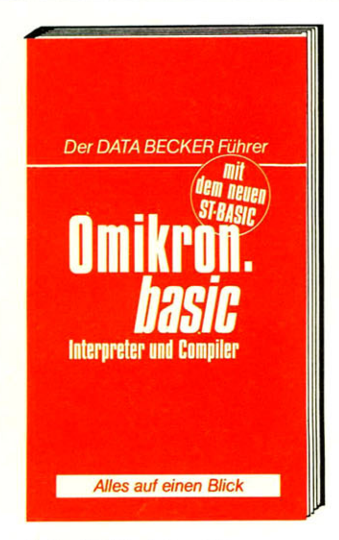
Listing geht weiter...





ICH.ST





OmikronBASIC auf einen Blick.
Der DATA BECKER Führer zu OmikronBASIC – das schnelle Nachschlagewerk für jeden Programmierer. Hier finden Sie schnell und zuverlässig alle Anweisungen und Funktionen mitsamt ihren Parametern. Übersichtlich geordnet nach Sachgruppen, alphabetisch mit Kurzsyntax und nach Stichworten. Für Ihre Programmierarbeit einfach unentbehrlich. Mit Interpreter- und Compiler-Version.
Der DATA BECKER Führer zu OmikronBASIC
235 Seiten, DM 24,80



Ihr unbedingt erstes Buch zum ST.
ST für Einsteiger – hier ist der Erfolg mit Ihrem neuen Rechner vorprogrammiert. Denn hier finden Sie alles Wichtige leichtverständlich und systematisch erklärt: der Anschluß des Rechners, Bedienung des GEM-Desktops, Kopieren von Disketten, Installieren von Festplatten und Druckern, die wichtigste Standardsoftware und natürlich eine Einführung in die Programmierung mit OMIKRON-BASIC. Egal mit welchem ST-Modell Sie arbeiten, dieses Buch macht Ihnen den Einstieg leicht.
ST für Einsteiger
ca. 250 Seiten, DM 29,–



Bessere Texte mit BECKERtext ST.
Wenn Sie all die Möglichkeiten von BECKERtext ST – besonders die der Version 2.0 – zu nutzen wissen, werden Sie nur ungern auf diese Textverarbeitung verzichten. Denn hier finden Sie alles, um ansprechenden, perfekt gestalteten Text zu erstellen. Wie, das zeigt Ihnen das große Buch zu BECKERtext ST. Schritt für Schritt lernen Sie hier anhand zahlreicher Beispiele den gesamten Leistungsumfang dieses Programms kennen. Natürlich mit jeder Menge nützlicher Tips zur täglichen Arbeit.
Das große Buch zu BECKERtext ST
Hardcover, ca. 300 Seiten
inkl. Diskette, DM 69,–



ADIMENS und ADITALK – kurz und knapp.
Der DATA BECKER Führer zu ADIMENS ST/GT – das umfassende Nachschlagewerk zu ADIMENS und ADITALK. Auf einen Blick finden Sie hier all die Informationen, die Sie für eine effektive Arbeit mit diesem Programm brauchen – zusammengestellt von zwei Autoren, die mit beiden Programmen bestens vertraut sind: Dieter und Jürgen Geiss, die Programmierer von ADIMENS und ADITALK.
Der DATA BECKER Führer zu ADIMENS
ca. 250 Seiten, DM 29,80



Gleich richtig einsteigen.
Lernen Sie den Leistungsumfang des neuen ST-BASIC gleich richtig kennen. Mit ST-BASIC für Einsteiger. Praxisorientiert und anhand zahlreicher Beispiele zeigt Ihnen der Autor, worauf Sie bei der Programmierung besonders achten müssen: Zuweisung und Variablendeklarationen, Ein- und Ausgaben, Schleifenprogrammierung, Diskettenoperationen, Grafikprogrammierung und und und. Und sollten Sie trotzdem mal nicht weiterwissen, eine Pannenhilfe sagt Ihnen, woran es liegt.

ST-BASIC für Einsteiger
ca. 250 Seiten, DM 29,–
erscheint ca. 11/88

COUPON

HIERMIT BESTELLE ICH FÜR MEINEN ATARI ST

NAME, VORNAME

STRASSE

ORT

zzgl. DM 5,– Versandkosten unabhängig von der bestellten Stückzahl
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

```
 94:
 95:  GET_NEXT: move.b    d0,d1   * letztes Byte sichern
 96:            move.b    (a0)+,d0 * nächstes Byte holen
 97:
 98:            dbra      d6,NEXT_BYTE  * Bytezähler
                                        dekrementieren
 99:
100:            bset      #END_SQZ,d5 * "Ende-der-
                                   Daten" Flag setzen
101:            bra.s     WRITE_SQZ      * und Daten
                                        schreiben
102:
103:
104:
105:  NEXT_BYTE:
106:            cmp.b     d0,d1 * vergl. aktuelles Byte
                                   mit Vorgänger
107:            bne.s     WRITE_SQZ
108:
109:            addq.w    #1,d3 * Zähler inkrementieren
110:
111:            bra.s     GET_NEXT * ... es gibt
                                     viel zu tun
112:  *                           ( packen wir's ein ! )
113:
114:
115:  *
116:  * Die Kette gleicher Bytes ist unterbrochen,
117:  * jetzt werden die Daten geschrieben:
118:
119:  WRITE_SQZ:
120:            cmp.w     d2,d3
                * Komprimierung lohnt erst ab >3 Bytes !
121:            bmi.s     WR_M_SINGLE
122:
123:
124:  *
125:  * Für mehr als 3 gleiche Bytes wird ein
126:  * Datenfeld (ESC)(Zähler)(Byte) geschrieben:
127:
128:            move.b    d5,(a1)+       * (ESC) Byte
                                        schreiben
129:
130:            cmp.w     d5,d3
131:            beq.s     WR_M_EOM
132:
133:  MULT_CT:  cmp.w     d4,d3
134:            bmi.s     M_NO_EOM
135:
136:            sub.w     d4,d3
137:
138:  M_CT_2:   move.b    #01,(a1)+ * (01) für
                                   jeweils 256 Bytes
139:            sub.w     d4,d3
140:            bpl.s     M_CT_2
141:
142:            add.w     d4,d3
143:
144:  WR_M_EOM: clr.b     (a1)+
145:
146:
147:
148:  M_NO_EOM: move.b    d3,(a1)+  * Zähler schreiben
149:            move.b    d1,(a1)+ * Datenbyte schreiben
150:            bra       RECOUNT
151:
152:
153:  *
154:  * Eine Komprimierung lohnt nicht,
155:  * die Datenbytes werden einzeln geschrieben:
156:
157:  WR_M_SINGLE:
158:            move.b    d1,(a1)+       * Datenbyte
159:            cmp.b     d5,d1
160:            bne.s     WR_M_NEQ
161:
162:
163:  *
164:  * Datenbyte = (ESC) Byte ?
165:  * jetzt muss (ESC)(ESC) geschrieben werden:
166:
167:            move.b    d5,(a1)+       * (ESC) Byte

168:
169:  WR_M_NEQ: dbra      d3,WR_M_SINGLE
170:            bra.s     RECOUNT
171:
172:
173:  *
174:  * Die Komprimierung ist beendet,
175:  * das zuletzt gelesene Byte muß
176:  * noch geschrieben werden:
177:
178:  END_SQUEEZE:
179:            move.b    d0,(a1)+ * letztes Datenbyte
180:            cmp.b     d5,d0
181:            bne.s     END_NEQ
182:
183:            move.b    d5,(a1)+* (ESC) Byte verdoppeln
184:
185:
186:  *
187:  * Die Datenmarke (ESC)(SAM)(00)
188:  * wird geschrieben, um das Ende der
189:  * Daten zu markieren:
190:
191:  END_NEQ:  move.b    d5,(a1)+             * (ESC)
192:            move.b    #$02,(a1)+           * (SAM)
193:            clr.b     (a1)+                * (00)
194:
195:  *
196:  * Teil IV.
197:  *
198:  * Jetzt wird geprüft, ob die komprimierten
199:  * Daten kürzer sind, als das Quellbild:
200:
201:            movea.l   DESTINATION(sp),a2
                        * Startaddress Zielgebiet
202:            suba.l    a2,a1  * Ende-Start
                                 = kompr. Länge
203:            move.l    a1,d0  * in d0 zurückliefern
204:
205:            cmpi.w    #PICTURELEN,d0 * Ergebnis >
                                        32000 Bytes ?
206:            ble.s     ITS_OK * nein, dann war's gut
207:
208:
209:  * Teil IVb.
210:  *
211:  * Das Ergebnis ist länger !!!
212:  * -> also wird einfach das Quellbild
213:  *    kopiert.
214:
215:            move.w    #PICTURELEN/4-1,d0  * Zähler
216:            movea.l   PICTURE(sp),a0 * Quellbild
217:            movea.l   a2,a1          * Zielgebiet
218:
219:            clr.b     (a1)+    * "segment_length" = 0
220:  *                              = unkomprimiert !
221:
222:
223:  * Bildinformation BYTEWEISE kopieren !
224:
225:  COP_LOOP: move.b    (a0)+,(a1)+    * block copy
226:            move.b    (a0)+,(a1)+
227:            move.b    (a0)+,(a1)+
228:            move.b    (a0)+,(a1)+
229:            dbra      d0,COP_LOOP
230:
231:            suba.l    a2,a1           * Länge
232:            move.l    a1,d0           *in d0 zurück
233:
234:  ITS_OK:   movea.l   (sp)+,a0    * Heimataddresse
235:            addq.l    #PICTURE,sp * stack korrigie.
236:            jmp       (a0)        * bye ...
237:
238:  *===============================================
239:  *
240:  * Escape - Byte ermitteln.
241:  * Das Escape - Byte ist das Byte, das
242:  * am wenigsten häufig im Bild vorkommt ...
243:  *
244:  * a0: Adresse Bilddaten
245:
246:  GET_ESCAPE:
247:            link      a6,#-512  * Zähltabelle
                                     einrichten
248:
249:
250:  * Teil I.
```

Listing geht weiter...

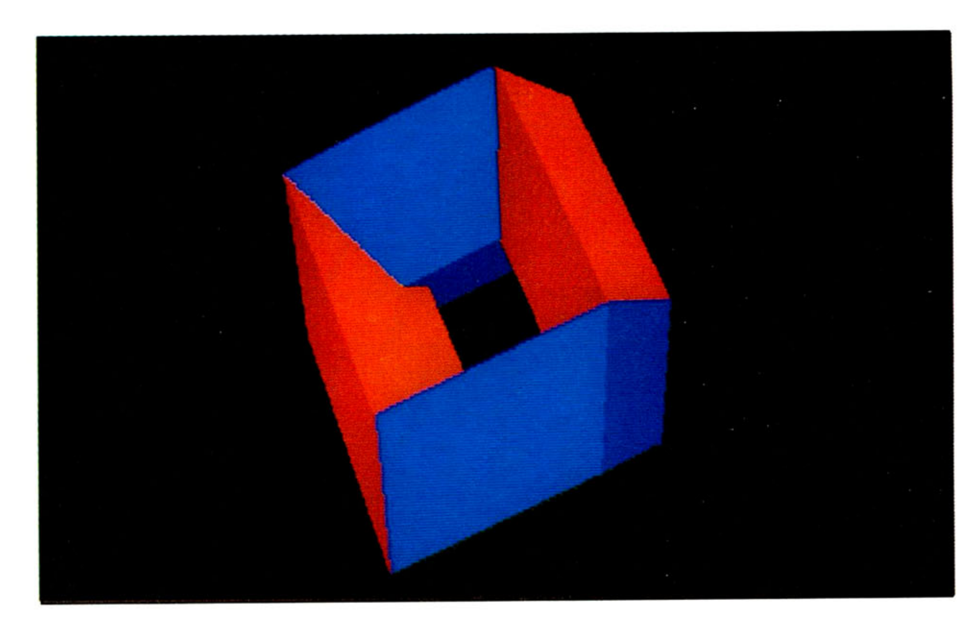
```
251:  *
252:  * Zähltabelle auf Null setzen:
253:
254:            movea.l   sp,a3 * Startaddresse Tabelle
255:            moveq     #127,d3   * 256 Zählerwerte
256:
257:  CLR_LOOP: clr.l     (a3)+     * Tabelle löschen
258:            dbra      d3,CLR_LOOP
259:
260:
261:  * Teil II.
262:  *
263:  * Byte - Häufigkeiten im Bild ermitteln:
264:
265:            movea.l   sp,a3
266:            move.w    #PICTURELEN-1,d4 * 32000
                                     Bytes zählen
267:
268:  ADD_LOOP: clr.w     d3
269:            move.b    (a0)+,d3        * Byte holen
270:            add.w     d3,d3
271:            addq.w    #1,0(a3,d3.w) *Zählwert für
                                jew. Byte erhöhen
272:
273:            dbra      d4,ADD_LOOP
274:
275:
276:  * Teil III.
277:  *
278:  * Byte mit der geringsten Häufigkeit
279:  * ermitteln:
280:
281:            move.w    #255-3,d3
282:            move.w    (a3),d4 * Startwert:Byte (00)
283:            addq.l    #6,a3   * Bytes (01) und (02)
                                  sind reserviert !
284:
285:                moveq      #0,d0   * Noch ist (00) der
                                          Favorit ...
286:                moveq      #$03,d2 * ... und wir fahren
                                          bei (03) fort
287:
288:    CMP_LOOP: move.w     (a3)+,d5
289:                beq.s      FINISH * Null ? besser geht's
                                          nicht
290:
291:                cmp.w      d4,d5   * besser ?
292:                bpl.s      MORE
293:
294:                move.w     d5,d4   * neuer Anhaltswert
295:                move.w     d2,d0   * und ein neuer Favorit
296:
297:    MORE:       addq.w     #1,d2   * nächster Kandidat
298:                dbra       d3,CMP_LOOP
299:
300:                bra.s      RETURN
301:
302:    FINISH:     move.w     d2,d0
303:
304:
305:
306:    RETURN:     unlk     a6        * stack freigeben
307:                rts                * Return To Somewhere
308:
309:    *==============================================
310:
311:                end
```

Listing 3: Das Einpackprogramm

ENDE

DIE PIPELINES IN CAD3D

CAD3D ist eines der wenigen 3D-CAD Programme, die es für den ATARI ST-Computer gibt. In dem Review in dieser Zeitung [3] wurde darauf verwiesen, daß der Autor Tom Hudson eine Communication Pipeline entworfen hat, die allen Programmierern Zugang zu den Routinen dieses Programms ermöglicht. Dieses ist nun das erste Beispiel, wie man sich die Routinen des CAD3D für eigene Zwecke zunutze macht. Die Originaldokumentation aller Routinen findet man auf der 3D-Developers Disk for CAD3D, Best. Nr. ST0244 der Firma Antic Publishing.



Die tägliche Neuerfindung des Rades

Nichts ist uneffektiver und frustrierender, als nach tage-, wochen- oder gar monatelanger Arbeit festzustellen, daß es für das Problem, das man gerade unter den Tasten hat, bereits eine Lösung (sprich: ein Programm) gibt. Eines der Probleme, die man getrost vergessen kann, ist die Darstellung von Körpern auf dem flachen Bildschirm des Computers; vorausgesetzt, man hat CAD3D und die nötige Zeit, das Objekt einzugeben. Hat man beide Hürden genommen, wird man durch eine gelungene 3D-Darstellung entschädigt, womöglich noch in Farbe und Stereo. Wer von CAD3D bisher nichts gehört hat, sollte seinen Stapel alter ST-Hefte nochmals durchwühlen und nach [3] suchen. Dort findet sich ein Bericht über dieses leistungsfähige 3D-Programm. Kommen wir nun zum Kern der Sache. Schon seit längerem spukt mir ein Artikel über den Hypercubus im Kopf herum, der in der **Zeitschrift Spektrum der Wissenschaft** [2] erschienen ist. Ein Hypercubus ist nichts anderes als ein vierdimensionaler Würfel. Und so, wie man einen dreidimensionalen Würfel auf ein Blatt Papier projizieren kann, kann man einen vierdimensionalen Würfel in den dreidimensionalen Raum (unsere Umwelt) projizieren. Das Ergebnis ist ein Körper, dessen Umrisse nicht immer denen eines Würfels ähneln. Ein erster Versuch, sich das Ergebnis vorzustellen, ist in [4] veröffentlicht. Hier allerdings ist der Körper als Kantenmodell dargestellt. Und das ist nicht zu vergleichen mit dem räumlichen Eindruck, den CAD3D ermöglicht. Und wenn noch eine Stereobrille für den rechten räumlichen Eindruck sorgt, dann steht einer experiementellen räumlichen Geometrie nur noch die eigene Phantasie im Weg.

Der Hypercubus

Abb.1 zeigt einen Würfel, dessen Eckpunkte um den Ursprung eines kartesischen Koordinatensystems herum angeordnet sind. Jeder Punkt in einem solchen dreidimensionalen Koordinatensystem ist durch ein Zahlentripel eindeutig festgelegt. Diese drei Zahlen heißen die Koordinaten des Punktes. Nehmen wir der Einfachheit halber an, daß die Kantenlänge des Würfels 2 beträgt, dann lassen sich die Koordinaten der 8 Eckpunkte des Würfels leicht angeben:

W_0	(-1/-1/-1)
W_1	(+1/-1/-1)
W_2	(-1/+1/-1)
W_3	(+1/+1/-1)
W_4	(-1/-1/+1)
W_5	(+1/-1/+1)
W_6	(-1/+1/+1)
W_7	(+1/+1/+1)

Diese Eckpunkte sind (soweit sichtbar) in Abb.1 mit ihrem Index eingezeichnet. Einen vierdimensionalen Hypercubus erhalten wir, indem wir uns in die vierte Dimension aufschwingen und das gleiche Bildungsgesetz für neu hinzukom-

mende Eckpunkte des Hypercubus anwenden:

H_0	(-1/-1/-1/-1)
H_1	(+1/-1/-1/-1)
H_2	(-1/+1/-1/-1)
H_3	(+1/+1/-1/-1)
H_4	(-1/-1/+1/-1)
H_5	(+1/-1/+1/-1)
H_6	(-1/+1/+1/-1)
H_7	(+1/+1/+1/-1)
H_8	(-1/-1/-1/+1)
H_9	(+1/-1/-1/+1)
H_{10}	(-1/+1/-1/+1)
H_{11}	(+1/+1/-1/+1)
H_{12}	(-1/-1/+1/+1)
H_{13}	(+1/-1/+1/+1)
H_{14}	(-1/+1/+1/+1)
H_{15}	(+1/+1/+1/+1)

So leicht die Eckpunkte dieses Hypercubus auch anzugeben sind, man kann sich dieses Gebilde nicht mehr vorstellen (oder können Sie etwa...???).

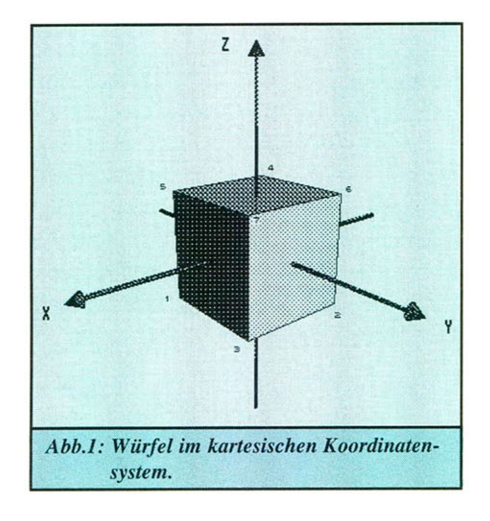


Abb.1: Würfel im kartesischen Koordinatensystem.

Schattenspiele

Stellen Sie sich vor, Sie nehmen den Würfel aus Abb.1 als Kantenmodell und stellen ihn in die Mittagssonne. Auf dem Boden sehen Sie dann einen Schatten. Der Mathematiker spricht von einer Projektion eines dreidimensionalen Körpers auf eine Ebene (zweidimensionale Fläche). Die Projektion eines Eckpunktes auf die xy-Ebene läßt sich mathematisch ganz einfach dadurch berechnen, daß man die z-Koordinate des entsprechenden Punktes 0 setzt. W0', der Schatten von W0 auf die xy-Ebene, hätte demnach die Koordinaten W0'(-1/-1/0). Bei dieser Projektion fallen dann jeweils die Projektionen zweier Eckpunkte zusammen; in diesem Fall W0' und W4'. Durch dieses Verfahren gelingt es uns, einen dreidimensionalen Körper auf einer zweidimensionalen Fläche darzustellen. Die Punkte des dreidimensionalen Raumes hatten wir in einem Koordinatensystem angegeben, in dem drei Achsen wechselseitig aufeinander senkrecht stehen und in gleiche Einheiten eingeteilt sind (kartesisches Koordinatensystem). Analog dazu denkt man sich ein kartesisches Koordinatensystem für den vierdimensionalen Raum so, daß die vier Koordinatenachsen ebenfalls aufeinander senkrecht stehen; wobei die vierte Achse auf den drei Achsen, die den dreidimensionalen Raum aufspannen, senkrecht steht, in diesem also nicht als Achse sichtbar ist. Falls Sie sich das noch räumlich vorstellen können, gehören Sie zweifellos zu der bevorzugten Gattung Mensch, die einen Rubick Cube aus jeder Stellung mit einer einzigen Drehung in der vierten Dimension innerhalb von 0.02 msec in seine Ausgangslage drehen kann. Aber rechnen kann man in der vierten Dimension vorzüglich. Vor allem lassen sich die Eckpunkte des Hypercubus aus der vierten Dimension in den dreidimensionalen Raum projizieren. Aber dieser "Schatten" ist nicht flächig sondern räumlich (eben dreidimensional). Auch bei dieser Projektion entlang der vierten Koordinatenachse (Parallelprojektion) fallen jeweils zwei Punkte zusammen, z.B. H0'(-1/-1/-1/0) und H8'(-1/-1/-1/0). Die Projektion des Hypercubus in den dreidimensionalen Raum sieht also wie ein normaler Würfel aus. Interessant wird es erst, wenn man den Hypercubus in einer Ebene, die die vierte Koordinatenachse enthält dreht. Dann nämlich werden durch die Drehung die beiden Punkte H0' und H8' im dreidimensionalen Raum getrennt und fallen nicht mehr zusammen.

Mathematische Pirouetten

Um die Drehung eines Körpers zu erklären, bedarf es zunächst der Einführung eines neuen Begriffes: des Vektors. Dies ist ein gerichtetes Objekt (Pfeil) mit einer bestimmten Länge, einer bestimmten Richtung im Raum (Trägergerade) und einer Orientierung. Bindet man nun einen Vektor am Ursprung des Koordinatensystems fest, erhält man einen Ortsvektor, dessen Spitze auf einen Punkt im Raum zeigt (Abb.2). Der Ortsvektor zum Punkt H0 hätte dann folgende Darstellung:

$$r_0 = \begin{matrix} -1 \\ -1 \\ -1 \\ -1 \end{matrix}$$

Im Gegensatz zur Koordinatenschreibweise eines Punktes, schreibt man den affinen (verwandten) Ortsvektor senkrecht (Spaltenschreibweise) um Verwechslungen zu vermeiden. Will man nun den Punkt um eine Koordinatenachse drehen, läßt sich das mit Hilfe der Vektorrechnung einfach als Multiplikation des Ortsvektors mit der Drehmatrix erledigen.

Die Wirbelmatratze (Drehmatrix)

Um die Drehung in einem vierdimensionalen Vektorraum auszuführen, braucht man eine 4x4-Matrix folgenden Aufbaus: Die Hauptdiagonale der Drehmatrix ist 1, außer an den Stellen, die den Index einer der Achsen der Drehebene tragen. Die Außendiagonalelemente sind alle Null, außer denjenigen, deren Index gleich einer Achsennummer der Drehebene ist. Die Werte der Diagonalelemente sind der Cosinus des Drehwinkels, die Werte der Außerdiagonalelemente sind der Sinus

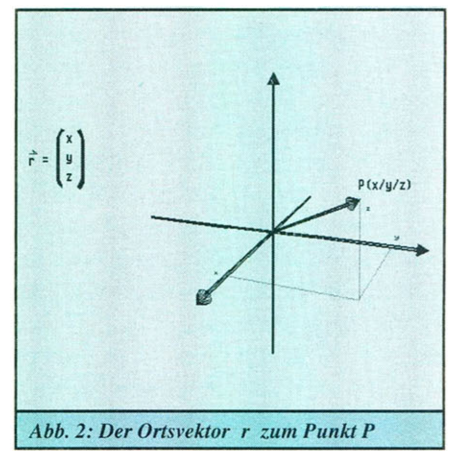


Abb. 2: Der Ortsvektor r zum Punkt P

bzw. Sinus des Drehwinkels, je nach Anordnung. Nehmen wir z.B. an, es soll eine Drehung in der 1-2-Ebene um den Winkel a durchgeführt werden. Die Drehmatrix hätte dann die Form:

1	0	0	0
0	cosa	sina	0
0	-sina	cosa	0
0	0	0	1

Die Koordinatenachsen sind der Einfachheit halber von 0 bis 3 durchnumeriert. Das Ergebnis der Drehung, also den gedrehten Ortsvektor, erhält man durch Multiplikation des Ortsvektors mit der Drehmatrix:

$$r_{j'} = \sum_{i=1}^{4} r_i * a_{ij} \; ; \quad 0<=j<=4$$

Wer sich eingehender mit der Mathematik beschäftigen möchte, dem sei [1] empfohlen.

Zur Sache CADzchen

Kommen wir zum Kern der Sache. CAD-3D erlaubt wie gesagt die Benutzung seiner Routinen von fremder Seite über sogenannte Pipelines. Hierzu muß das Anwenderprogramm gleichzeitig mit dem CAD-3D im Speicher resident sein. Das ist am einfachsten zu erreichen, wenn das Anwenderprogramm als Accessory vorliegt. Um dem Hypercubus in den dreidimensionalen Teil seines vierdimensionalen Angesichts zu blicken, habe ich ein Accessory geschrieben, das die Koordinaten des Hypercubus berechnet, eine Drehung in einer der Ebenen der vierten Dimension ermöglicht und schließlich ein Objekt erzeugt, welches von CAD-3D verarbeitet werden kann und der Projektion des vierdimensionalen Hypercubus in den dreidimensionalen Objektraum von CAD-3D entspricht. Das so erzeugte Objekt wird an CAD-3D über die Communication-Pipeline übergeben und steht anschließend für weitere Manipulationen in CAD-3D zur Verfügung. Um den Anblick möglichst anschaulich zu gestalten, habe ich einen Trick angewendet. Schaut man sich die Punkte H0 bis H15 des Hypercubus genauer an, so erkennt man 8 Würfel mit jeweils einer gemeinsamen Komponente. Beispielsweise der Würfel bestehend aus H0 bis H7, bei denen die 4. Komponente +1 ist. Diesen Würfel nennt das Programm cubus0. Cubus1 ist dann der Teilwürfel, dessen Eckpunkte die 3. Komponente +1 besitzen etc. Cubus1 und cubus7 bilden somit zwei Würfel in CAD-3D, die sich decken. Erst bei einer Drehung in einer Ebene außerhalb des dreidimensionalen Raumes entfällt diese Entartung. Wer einen Farbmonitor besitzt, mag die verschiedenen Teilwürfel nach der Erzeugung beliebig einfärben. Das erhöht bei der anschließenden Betrachtung die Übersichtlichkeit. Geradezu ergreifend wirkt natürlich eine Farbdarstellung mit der Stereobrille.

Die Communication Pipeline

In CAD-3D werden, wie in allen Programmen, die GEM intensiv nutzen, die einzelnen Subroutinen über das event handling verwaltet. Der Autor von CAD-3D ist allerdings hier sehr systematisch vorgegangen und hat jedem Unterprogramm eine Messagenummer mit eindeutig definierten Parametern gegeben, so daß auch von externen Programmen, die gleichzeitig im Speicher sind (Accessories), auf die CAD-3D Routinen zugegriffen werden kann. Die Nummern der einzelnen Routinen und die benötigten Parameter findet man in der Dokumentation zur CAD-3D Developers Disk, die bei ANTIC Software erhältlich ist und für Programmierer eine notwendige Anschaffung ist, wenn man eigene Applikationen für CAD-3D schreiben will. Kennt man die Message Nummern, dann sieht ein typischer Aufruf einer CAD-3D Funktion so aus:

1) Schreibe eine Message mit den zugehörigen Parametern in den Message Puffer.
2) Schicke diese Nachricht an CAD-3D.
3) Warte bis CAD-3D eine Nachricht zurückschickt.
4) Schaue in der Nachricht von CAD-3D nach, ob die gewünschte Operation erfolgreich verlaufen ist. Wenn nicht, beende das Accessory.

Es gibt bereits ein Motion Control Accessory in der PD von ANTIC Software, das diese Pipeline ausnutzt. Bessere Versionen sollen folgen. Der Hypercubus jedenfalls nutzt diese Pipeline aus, um die Projektionen der Würfel cubus0 bis cubus7 in das CAD-3D-Programm einzuspeisen. Dazu werden zunächst die Objekte **cubus0 - cubus7** gelöscht, falls sie bereits vorhanden sind. Dann werden die Würfelkoordinaten an die CAD-3D Objektstruktur übergeben und die Funktion zur Übernahme der Struktur aufgerufen. Wenn dieser Aufruf erfolgreich verläuft, verabschiedet sich das Accessory, und auf dem Bildschirm des CAD-3D Programms stehen die 8 Teilwürfel des Hypercubus für weitere Manipulationen zur Verfügung.

Fazit:

Abb.4 zeigt die Teilwürfel **cubus0, cubus3, cubus4 und cubus7** des Hypercubus in Farbe, wenn er in den Ebenen 0-3,

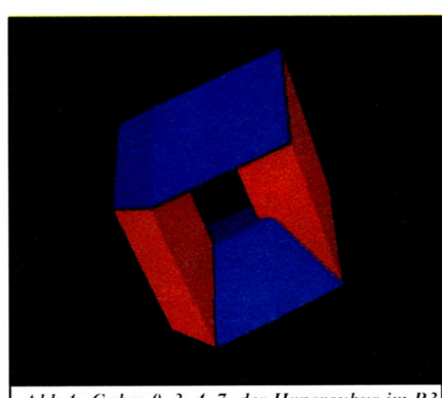


Abb.4: Cubes 0, 3, 4, 7 des Hypercubus im R3 Alle Drehwinkel 45°

1-3, 2-3 jeweils um 45 ° gedreht wurde. Listing 1 zeigt das C-Listing des Accessories. Es wurde mit dem Megamax C-Compiler entwickelt. Bitte haben Sie Verständnis dafür, daß wir die RSC-Datei nicht mitabdrucken, da dies den RAhmen deutlich sprengen würde. Wer diese trotzdem gerne zur Verfügung haben möchte, kann sie auf unserer Monatsdiskette November / Dezember ´88 bestellen.

Dr. Karl Sarnow

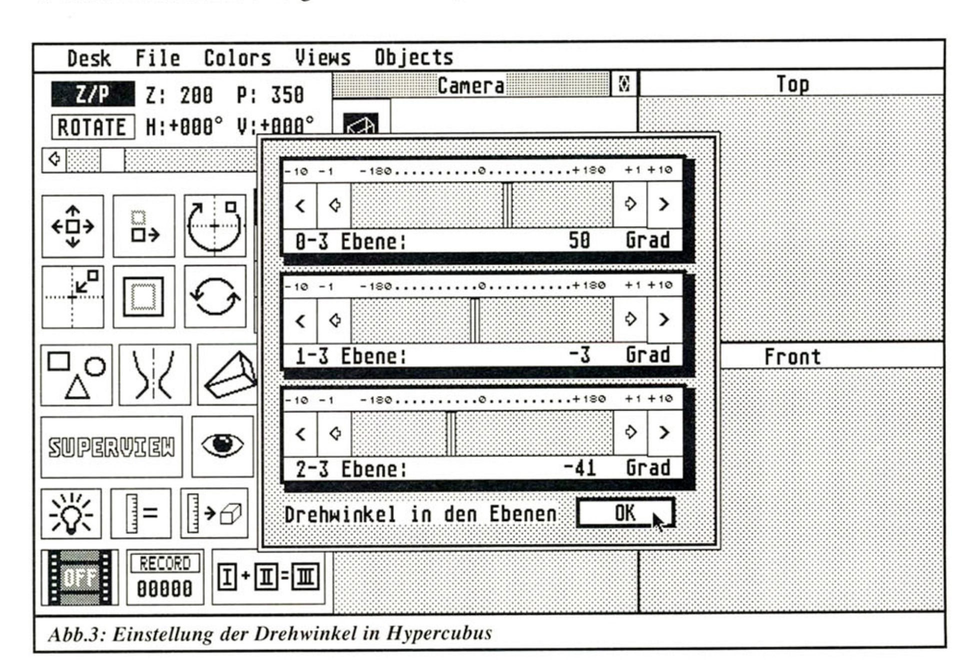
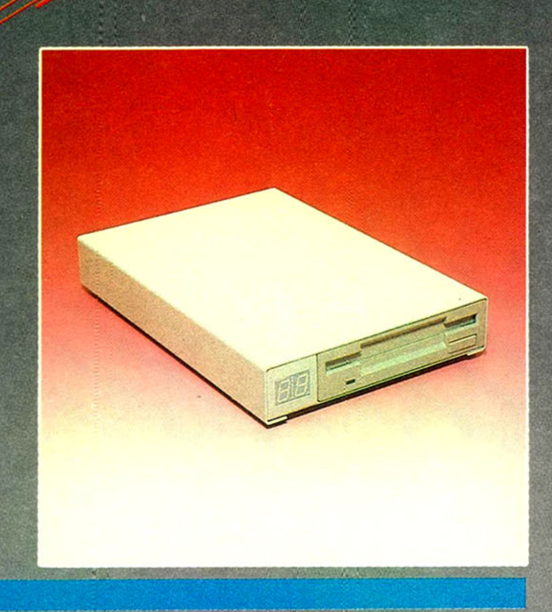


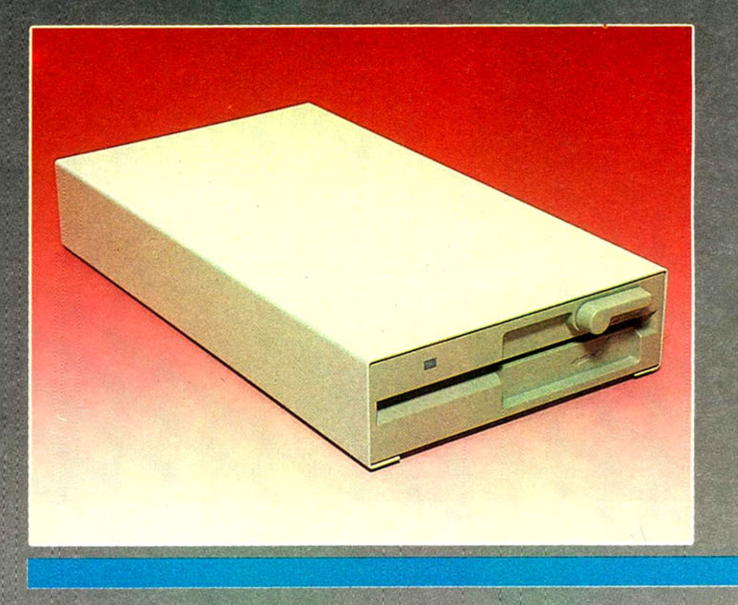
Abb.3: Einstellung der Drehwinkel in Hypercubus

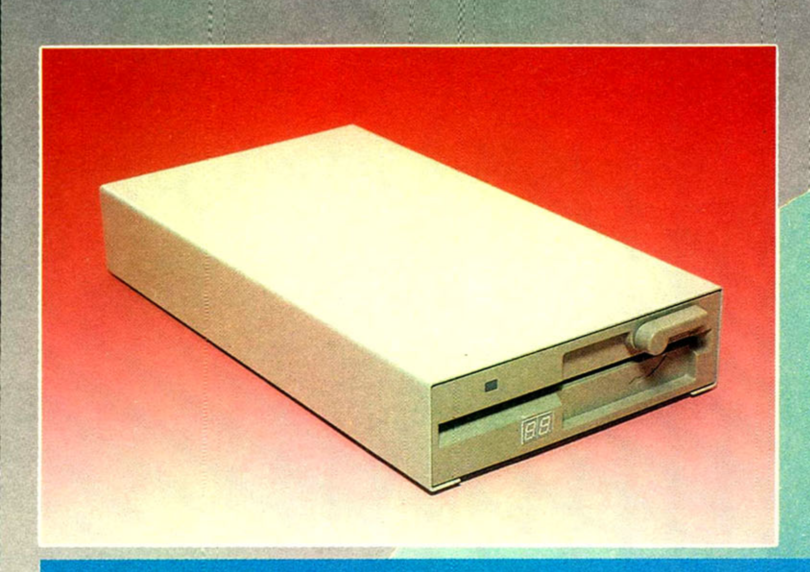


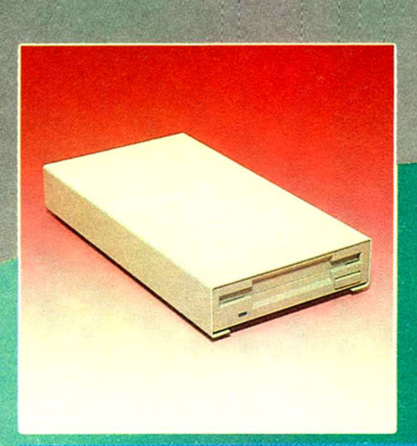
☎ 02 31/81 83 25-27
Telefax 02 31/81 74 29
D-4600 Dortmund 1
Burgweg 52 a











Wir liefern im 3-Tage-Rhythmus

Golem für Atari

Qualitätslaufwerke von einem der führenden Hersteller für Computerperipherie
— jetzt auch für alle Atari —

Golem Drive 3,5 Display
NEC 1037a mit heller Frontblende ● Atari-farbenes Metallghäuse ● Abschalter ● Trackdisplay zur aktuellen Spur- und Kopfanzeige **DM 359.-**

Golem Drive 3,5
NEC 1037a mit heller Frontblende ● Atari-farbenes Metallgehäuse ● Abschalter **DM 339.-**

Golem Drive 5,25 Display
NEC Laufwerk mit heller Frontblende ● Atari-farbenes Metallgehäuse ● Abschalter ● 40/80 Track-Umschalter Trackdisplay zur aktuellen Spur- und Kopfanzeige **DM 449.-**

Golem Drive 5,25
NEC Laufwek mit heller Frontblende ● Atari-farbenes Metallgeäuse ● Abschalter ● Abschalter ● 40/80 Track-Umschalter **DM 419.-**

Literatur:

[1] Dewdney, A.K. Hyperkubus. Sonderheft Spektrum der Wissenschaft. Computer Kurzweil. p.25ff.

[2] Kowalsky, H.J. Lineare Algebra. Göschens Lehrbücherei, Band 27.

[3] Sarnow, K. CAD-3D - die dritte Dimension. ST Computer 11/87.

[4] Sarnow, K. Der Hypercubus. c't 2/87. p.124.


```
  1:  #include <stdio.h>
  2:  #include <gembind.h>
  3:  #include <gemdefs.h>
  4:  #include <obdefs.h>
  5:  #include <osbind.h>
  6:  #include <math.h>
  7:  #include <vt52.h>
  8:  #include "hypercub.h" /*Entfällt bei Einbindung
                                 von Listing 2 oder 3*/
  9:
 10:  #define TRUE (1)
 11:  #define FALSE (0)
 12:
 13:  extern int gl_apid;
 14:  int menu_id,cad_id,forever;
 15:
 16:  int contrl[12],
 17:      intin[128],
 18:      intout[128],
 19:      ptsin[128],
 20:      ptsout[128] ;
 21:
 22:  int work_in[12],
 23:      work_out[57] ;
 24:
 25:  int pxyarray[10] ;
 26:
 27:  int handle ;
 28:
 29:  OBJECT *regler_tree; /* Das Formular in Abb.3 */
 30:
 31:  int mouse ;           /* mouse ein = 1, aus = 0 */
 32:  int gl_dummy ;        /* globaler dummy */
 33:
 34:  int msgbuff[16] ;
 35:  int whand ;
 36:
 37:  int xfull,yfull,wfull,hfull; /* Fensterkoord. */
 38:  int x_regl,y_regl,w_regl,h_regl;
      /* Koordinaten des Formulars */
 39:
 40:  char *text[3]; /* Zeiger a.Strings i.TEDINFO */
 41:  double winkel,zentr,kubus[16][4],
             dreh[4][4],ecke[16][4];
 42:
 43:  int cube[8]={-1,-1,-1,-1,-1,-1,-1,-1};
 44:  /* ID's der Würfel */
 45:
 46:  int vertx[8][8],verty[8][8],vertz[8][8];
 47:  /* 8 Punkte der8 Würfel */
 48:
 49:  int verta[12]= /*Die Punkte eines Würfels*/
 50:  {5,5,7,7,6,6,4,4,5,5, 3, 1};  /*bilden Dreiecks-
                                     flächen. Im Feld
                                     verta sind */
 51:  /*alle Punkte A der 12 Dreiecke enthalten */
 52:  int vertb[12]= /*i.Feld vertb alle Punkte B
                      der 12 */
 53:  {1,3,3,2,2,0,0,1,7,6, 1, 0};  /* Dreiecksflächen
                                      und in vertc
                                      alle Punkte C */
 54:
 55:  int vertc[12]=
 56:  {3,7,2,6,0,4,1,5,6,4, 2, 2};
 57:
 58:  int color[12]=
 59:{0x060f,0x030f,0x060f,0x030f,0x060f,0x030f,0x060f,
   0x030f,0x060f,0x030f,0x050f,
 60:  0x060f};
 61:  /* Das niedere Byte enthält die Farbinformation
        (15)*/
 62:  /* Das obere Byte enthält die Information,
        welche Kante sichtbar ist*/
 63:
 64:  int pfad[64]=
 65:        {0,1,2,3,4,5,6,7,0,1,2,3,8,9,10,11,0,1,4,
           5,8,9,12,13,0,2,4,6,8,10,12,14,

 66:        1,3,5,7,9,11,13,15,2,3,6,7,10,11,14,15,4,
           5,6,7,12,13,14,15,8,9,10,11,12,13,14,15};
 67:  /* Welche Eckpunkte gehören zu welchen
        Teilwürfeln? */
 68:
 69:  int mgin[8];/* Event Message Puffer
                 f.CAD-3D Messages*/
 70:
 71:  /* Object load/unload structure */
 72:
 73:  struct obj_str
 74:  {
 75:  char os_name[9];
 76:  int os_points;
 77:  int os_faces;
 78:  int *os_x;
 79:  int *os_y;
 80:  int *os_z;
 81:  int *os_a;
 82:  int *os_b;
 83:  int *os_c;
 84:  int *os_color;
 85:  };
 86:  struct obj_str obstrc[8];
 87:
 88:  /* Buffer address structure */
 89:
 90:  char noCAD[]=
 91:  "[1][CAD-3D is not present!][ OK ]";
 92:
 93:
 94:  /*************************************/
 95:
 96:
 97:  open_vwork()
 98:  {
 99:     int i,dummy;
100:
101:     appl_init();
102:     handle=graf_handle(&gl_dummy,&gl_dummy,
                      &gl_dummy,&gl_dummy);
103:     menu_id=menu_register(gl_apid," Hypercubus");
         /*Hypercubus in die Accessory-Leiste */
104:  }  /*einklinken*/
105:
106:  close_vwork()   /* Ausstieg, wenn Fehler */
107:  {
108:   appl_exit();
109:   exit();
110:  }
111:
112:  show_mouse()    /* Weg mit der Maus */
113:  { if (!mouse)
114:    { graf_mouse(257,&gl_dummy) ;
115:      mouse = TRUE ;
116:    }
117:  }
118:
119:  hide_mouse()    /* Und wieder her damit */
120:  { if (mouse)
121:    { graf_mouse(256,&gl_dummy) ;
122:      mouse = FALSE ;
123:    }
124:  }
125:
```

Listing geht weiter..

```
126:  rotation(temp)   /* Drehung d.Hypercubus durch */
127:  register double (*temp)[16][4]; /* Multiplikation
                                        der Ortsvektoren */
128:  {                                /* mit der Drehmatrix */
129:     register int j,k,l;
130:
131:     for (j=0;j<16;j++)
132:        for (k=0;k<4;k++)
133:        {
134:           (*temp)[j][k]=0;
135:           for (l=0;l<4;l++)
                 (*temp)[j][k]+=kubus[j][l]
                 *dreh[k][l];
136:        }
137:  }
138:
139:
140:  multi()
141:  { int dummy,i1,i2,achse1,achse2 ;
142:
143:
144:     wind_get(0,WF_WORKXYWH,&xfull,&yfull,&wfull,
              &hfull);
     /* Hol die Fensterkoordinaten */
145:     whand=wind_create(0,xfull,yfull,wfull,hfull);
     /* Öffne Fenster */
146:     wind_open(whand,xfull,yfull,wfull,hfull);
147:     do_regler();      /* Formular abarbeiten */
148:     for (i1=0;i1<16;i1++)
149:        for (i2=0;i2<4;i2++) kubus[i1][i2]=
           ((i1 & (1 << i2))==0) ? -1.0 : 1.0;
           /* Erzeuge Koordinaten */
150:     for (dummy=0;dummy<3;dummy++)
151:        {
152:          switch(dummy)
153:          {
154:             case 0:
155:               achse1=0;
156:               achse2=3;
157:                 break;
158:
159:             case 1:
160:               achse1=1;
161:               achse2=3;
162:                 break;
163:
164:             case 2:
165:               achse1=2;
166:               achse2=3;
167:                 break;
168:
169:          } /* Fallunterscheidung d.3 Drehebenen */
170:          for (i1=0;i1<4;i1++)
171:          for (i2=0;i2<4;i2++)
172:             {
173:                if (i1==i2) dreh[i1][i2]=1;
                /* Initialisierung d.Drehmatrix */
174:                else dreh[i1][i2]=0;
175:             }
176:             winkel=atol(text[dummy]);/* Winkel aus
                                     TEDINFO Struktur lesen
*/
177:             dreh[achse1][achse1]=dreh[achse2]
             [achse2]=cos(winkel*dcpi/180);
             /* Diagonalelemente setzen */
178:             dreh[achse1][achse2]=sin(winkel*dcpi/
180);
             /* AuPerdiagonalelemente setzen */
179:             dreh[achse2][achse1]=-
dreh[achse1][achse2];
180:             rotation(&ecke); /*Drehung ausführen*/
181:             for (i1=0;i1<16;i1++)/* Ergebnis
                                         umspeichern */
182:             for (i2=0;i2<4;i2++)
                 kubus[i1][i2]=ecke[i1][i2];
183:        }
184:         do_acc(); /* Übergabe an CAD-3D */
185:  } /* multi */
186:
187:  schieben(nr,vater,kind) /* Manage d.Bedienung
                                des Schiebers */
188:  int nr,vater,kind;
189:  {
190:     int mx,my,button,state,position;
191:     long zwischen;
192:
193:     evnt_button(1,1,1,&mx,&my,&button,&state);
194:     position=graf_slidebox(regler_tree,vater,
                                    kind,0);
     /* Finde Reglerposition 0-1000 beim
        Loslassen */
195:     zwischen=((long)
              (regler_tree[vater].ob_width)-(long)
              (regler_tree[kind].ob_width))
              *((long)(position))/1000L;
     /* Berechnung der neuen Reglerposition */
196:     regler_tree[kind].ob_x=(int)zwischen;
     /* Abspeichern der neuen Reglerposition */
197:     zwischen=((360L*(long)(position))/1000L)-180L;
     /* Umrechnung d.Schieberposition in Gradzahl*/
198:    sprintf(text[nr],"%ld",zwischen);
    /* Absp.d.Gradzahl als String in TEDINFO */
199:  }
200:
201:
202:  minus10(nr,vater,kind) /* Winkel um 10 Grad
                                 verringern */
203:  int nr,vater,kind;
204:  {
205:     long zwischen;
206:
207:     zwischen=atol(text[nr])-10L;
208:     if(zwischen<-180) zwischen=-180L;
209:     sprintf(text[nr],"%ld",zwischen);
210:     zwischen=((long)(regler_tree[vater].ob_width)
            -(long)(regler_tree[kind].ob_width))
            *(zwischen+180)/360L;
211:     regler_tree[kind].ob_x=(int)zwischen;
212:  }
213:
214:  minus1(nr,vater,kind)/* Winkel um 1° verringern*/
215:  int nr,vater,kind;
216:  {
217:     long zwischen;
218:
219:     zwischen=atol(text[nr])-1L;
220:     if(zwischen<-180) zwischen=-180L;
221:     sprintf(text[nr],"%ld",zwischen);
222:   zwischen=((long)(regler_tree[vater].ob_width)
            -(long)(regler_tree[kind].ob_width))
            *(zwischen+180)/360L;
223:     regler_tree[kind].ob_x=(int)zwischen;
224:  }
225:
226:  plus10(nr,vater,kind) /* Winkel um 10° erhöhen */
227:  int nr,vater,kind;
228:  {
229:     long zwischen;
230:
231:     zwischen=atol(text[nr])+10L;
232:     if(zwischen>180) zwischen=180L;
233:     sprintf(text[nr],"%ld",zwischen);
234:   zwischen=((long)(regler_tree[vater].ob_width)
            -(long)(regler_tree[kind].ob_width))
            *(zwischen+180)/360L;
235:     regler_tree[kind].ob_x=(int)zwischen;
236:  }
237:
238:  plus1(nr,vater,kind) /* Winkel um 1° erhöhen */
239:  int nr,vater,kind;
240:  {
241:     long zwischen;
242:
243:     zwischen=atol(text[nr])+1L;
244:     if(zwischen>180) zwischen=180L;
245:     sprintf(text[nr],"%ld",zwischen);
246:   zwischen=((long)(regler_tree[vater].ob_width)
            -(long)(regler_tree[kind].ob_width))
            *(zwischen+180)/360L;
247:     regler_tree[kind].ob_x=(int)zwischen;
248:  }
249:
250:  do_regler()     /* Manage das Formular */
251:  {
252:     int keycode;
253:     long zwischen;
254:
255:    form_dial(FMD_START,320,200,0,0,x_regl,y_regl,
    w_regl,h_regl); /* Zeige das Formular */
256:    form_dial(FMD_GROW,320,200,0,0,x_regl,y_regl,
    w_regl,h_regl); /* für die Wahl der Winkel */
```

Listing geht weiter...

```
257:      objc_draw(regler_tree,0,80,x_regl,y_regl,w_regl,
          h_regl);
258:      objc_change(regler_tree,REGLEX,0,x_regl,y_regl,
          w_regl,h_regl,NORMAL,1);
259:       do
260:          {
261:             keycode=form_do(regler_tree,0);
262:             switch(keycode)
263:                {
264:                 case MIN103:
265:                 minus10(0,VATER3,SCHIEBE3);
266:                 objc_draw(regler_tree,BOX3,
                               MAX_DEPTH,x_regl,y_regl,
                               w_regl,h_regl);
267:                 break;
268:
269:                 case MIN105:
270:                 minus10(1,VATER5,SCHIEBE5);
271:                 objc_draw(regler_tree,BOX5,
                               MAX_DEPTH,x_regl,y_regl,
                               w_regl,h_regl);
272:                 break;
273:
274:                  case MIN106:
275:                  minus10(2,VATER6,SCHIEBE6);
276:                  objc_draw(regler_tree,BOX6,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
277:                  break;
278:
279:                  case MIN13:
280:                  minus1(0,VATER3,SCHIEBE3);
281:                  objc_draw(regler_tree,BOX3,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
282:                  break;
283:
284:                  case MIN15:
285:                  minus1(1,VATER5,SCHIEBE5);
286:                  objc_draw(regler_tree,BOX5,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
287:                  break;
288:
289:                  case MIN16:
290:                  minus1(2,VATER6,SCHIEBE6);
291:                  objc_draw(regler_tree,BOX6,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
293:
294:                  case PLUS103:
295:                  plus10(0,VATER3,SCHIEBE3);
296:                  objc_draw(regler_tree,BOX3,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
297:                  break;
298:
299:                  case PLUS105:
300:                  plus10(1,VATER5,SCHIEBE5);
301:                  objc_draw(regler_tree,BOX5,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
302:                  break;
303:
304:                  case PLUS106:
305:                  plus10(2,VATER6,SCHIEBE6);
306:                  objc_draw(regler_tree,BOX6,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
307:                  break;
308:
309:                  case PLUS13:
310:                  plus1(0,VATER3,SCHIEBE3);
311:                  objc_draw(regler_tree,BOX3,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
312:                  break;
313:
314:                  case PLUS15:
315:                  plus1(1,VATER5,SCHIEBE5);
316:                  objc_draw(regler_tree,BOX5,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
317:                  break;
318:
319:                  case PLUS16:
320:                  plus1(2,VATER6,SCHIEBE6);
321:                  objc_draw(regler_tree,BOX6,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
322:                  break;
323:
324:                  case SCHIEBE3:
325:                  schieben(0,VATER3,SCHIEBE3);
326:                  objc_draw(regler_tree,BOX3,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
328:
329:                  case SCHIEBE5:
330:                  schieben(1,VATER5,SCHIEBE5);
331:                  objc_draw(regler_tree,BOX5,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
332:                  break;
333:
334:                  case SCHIEBE6:
335:                  schieben(2,VATER6,SCHIEBE6);
336:                  objc_draw(regler_tree,BOX6,
                                MAX_DEPTH,x_regl,y_regl,
                                w_regl,h_regl);
337:                  break;
338:               }
339:          }
340:          while (keycode!=REGLEX);
341:          form_dial(FMD_SHRINK,320,200,0,0,x_regl,
                    y_regl,w_regl,h_regl);
342:          form_dial(FMD_FINISH,320,200,0,0,x_regl,
                    y_regl,w_regl,h_regl);
343:  }
344:
345:
346:  main()
347:  {
348:     int i,event;
349:
350:     open_vwork();
351:     if (Getrez()==2)
352:     {
353:        if (!rsrc_load("hypercub.rsc"))
354:        {
355:           form_alert(1,"[3][Ich kann HYPERCUB.RSC
                       nicht finden !][Abbruch]");
356:           close_vwork();
357:        }
358:     }
359:     else
360:     {
361:        if (!rsrc_load("hypercol.rsc"))
362:        {
363:           form_alert(1,"[3][Ich kann HYPERCOL.RSC
                       nicht finden !][Abbruch]");
364:           close_vwork();
365:        }
366:     }
367:     if (!rsrc_gaddr(R_TREE,REGLER,&regler_tree))
368:     {
369:        form_alert(1,"[3][Resource File nicht ok !]
                   [Abbruch]");
370:        close_vwork();
371:     }
372:     form_center(regler_tree,&x_regl,&y_regl,
                &w_regl,&h_regl);
373:     text[0]=(((TEDINFO *)
               (regler_tree[GRAD3].ob_spec))
               ->te_ptext);/* Zeiger auf
                            Formulartexte */
374:     text[1]=(((TEDINFO *)
               (regler_tree[GRAD5].ob_spec))
               ->te_ptext); /* initialisieren */
375:     text[2]=(((TEDINFO *)
               (regler_tree[GRAD6].ob_spec))
               ->te_ptext);
376:     for (i=0;i<3;i++) sprintf(text[i],"%d",0);
         /* Winkel auf 0 Grad setzen */
377:     while(forever==0)    /* Warten auf ... */
378:     {
379:        evnt_mesag(msgbuff);
380:        if(msgbuff[0]==40 && msgbuff[4]==menu_id)
            /* HYPERCUBUS wurde aufgerufen */
381:        {
382:           cad_id=appl_find("CAD3D2  ");
              /* Ist CAD-3D vorhanden? */
```

Listing geht weiter...

```
383:            if(cad_id<0)
384:              form_alert(1,noCAD); /* Nein,
                                          Abbruch */
385:            else
386:            {  /* Frage CAD-3d Nach RAM-STATUS */
387:             toCAD(0x3d00,0,0,0,0,0); /* Falls
               nach 2000 Einheiten keine Reaktion */
388:             /* liegt ein Fehler vor */
389:             is_there:
390:             event=evnt_multi(0x0030,-1,-1,-1,
391:                              0,0,0,0,0,
392:                              0,0,0,0,0,
                                  mgin,2000,0,
393:                             &gl_dummy,&gl_dummy,
                                 &gl_dummy,&gl_dummy,
                                 &gl_dummy,&gl_dummy);
394:             if(event & 0x0020)
395:               form_alert(1,noCAD);
396:             else
397:             {
398:               if(mgin[0]==0x3d80)  /* CAD-3D
                                   meldet Erfolg */
399:               multi();  /*Stelle Winkel ein und
                             Berechne Koord. */
400:               else
401:                 goto is_there;
402:             }
403:          }
404:        }
405:      }
406:   }
407:
408:   /* Found CAD-3D 2.0, let's rock-n-roll! */
409:
410:   do_acc()
411:   {
412:         register int ix,iy;
413:           int loeschen[40];
414:
415:   for (ix=0;ix<40;ix++) loeschen[ix]=0;
416:   for (ix=0;ix<8;ix++) /* Übergebe die berechneten
                             Koordinaten in der */
417:   {                    /* CAD-3D Object Struktur */

418:           sprintf(obstrc[ix].os_name,"cube%1d",ix);
419:           obstrc[ix].os_points=8;
420:           obstrc[ix].os_faces=12;
421:           obstrc[ix].os_x=vertx[ix];
422:           obstrc[ix].os_y=verty[ix];
423:           obstrc[ix].os_z=vertz[ix];
424:           obstrc[ix].os_a=verta;
425:           obstrc[ix].os_b=vertb;
426:           obstrc[ix].os_c=vertc;
427:           obstrc[ix].os_color=color;
428:           if (cube[ix]>=0) loeschen[ix]=TRUE;
429:   }
430:
431:   if (loeschen[0]) /*Loesche cubus0 - cubus7 falls
                          schon vorhanden*/
432:   {
433:       comm_it(mgin,0x3d07,loeschen,0,0,0);
       /* Löschbefehl an CAD-3D */
434:           if (mgin[3]<0)
435:           {
436:                   disp_err("error erasing object");
437:                   goto acc_exit;
438:           }
439:   }
440:
441:   for (ix=0;ix<8;ix++)
442:   {
443:       for (iy=0;iy<8;iy++) /*Eingabe
                          d.Punktkoordinaten */
444:         {              /* in die Objektstruktur */

445:           vertx[ix][iy]=1000
               *kubus[pfad[iy+ix*8]][0];
446:           verty[ix][iy]=1000
               *kubus[pfad[iy+ix*8]][1];
447:           vertz[ix][iy]=1000
               *kubus[pfad[iy+ix*8]][2];
448:         }
449:
450:         comm_it(mgin,0x3d05,&obstrc[ix],0,0,0);
           /* Befehl zur Übernahme des Objektes */
451:           if (mgin[3]<0) /* aus dem RAM */
452:           {
453:                   disp_err("error loading
                                  object");
454:                   goto acc_exit;
455:           }
456:
457:           cube[ix]=mgin[3]; /* Speichern des
                                     Objekt ID */
458:           comm_it(mgin,0x3d58,cube[ix],1,0,0);
               /* Zeige das Objekt */
459:           if (mgin[3]<0)
460:           {
461:                   disp_err("error selecting
                                  object");
462:                   goto acc_exit;
463:           }
464:   }
465:   /* Accessory is done, close/delete window and
          return to main routine */
466:
467:   hide_mouse();
468:
469:   acc_exit:
470:   wind_close(whand);
471:   wind_delete(whand);
472:   }
473:
474:   comm_it(rbuf,command,v1,v2,v3,v4,v5)
       /* Sende den Befehl an CAD-3D */
475:   int rbuf[],command,v1,v2,v3,v4,v5;
       /* und warte auf Antwort */
476:   {
477:   toCAD(command,v1,v2,v3,v4,v5);
478:
479:   do
480:    {
481:    evnt_mesag(rbuf);
482:    }
483:    while(rbuf[0]!=(command | 0x0080));
484:   }
485:
486:   /****************************************/
487:   /* Send standard message to CAD-3D 2.0 */
488:   /****************************************/
489:
490:   toCAD(command,v1,v2,v3,v4,v5)
491:   int command,v1,v2,v3,v4,v5;
492:   {
493:   msgbuff[0]=command;
494:   msgbuff[1]=gl_apid;
495:   msgbuff[2]=0;
496:   msgbuff[3]=v1;
497:   msgbuff[4]=v2;
498:   msgbuff[5]=v3;
499:   msgbuff[6]=v4;
500:   msgbuff[7]=v5;
501:   appl_write(cad_id,16,msgbuff);
502:   }
503:
504:
505:   /*********************************************/
506:   /* Display an error message in an alert box */
507:   /*********************************************/
508:
509:   disp_err(string)
510:   char string[];
511:   {
512:   char errstr[80];
513:
514:   strcpy(errstr,"[1][");
515:   strcat(errstr,string);
516:   strcat(errstr,"][ OK ]");
517:
518:   graf_mouse(0,&gl_dummy);
519:
520:   form_alert(1,errstr);
521:   }
```

Listing 1 : Das C - Listing des Accessories

Frankfurter Buchmesse '88



Auf der Frankfurter Buchmesse war der ATARI ST gut im Buchangebot vertreten. Es gab allerdings wenige Neuerscheinungen, für viele Verlage ist das Programm zum ST komplett.

Die Frankfurter Buchmesse 1988 vom 5. bis 10. Oktober war wieder einmal die größte aller Zeiten. Zum 40. Mal trafen sich Verlage, Autoren und Leser in Frankfurt, um diesmal 102 860 Neuerscheinungen bei fast 8 000 Ausstellern zu erleben. Insgesamt wurde laut Ausstellerstatistik die unvorstellbare Menge von 338 848 Titeln präsentiert.

Von besonderem Interesse für den ST-Benutzer war der zweite Stock der Halle 6, in dem die Fachverlage für Computertechnik zusammengefaßt waren. Das Angebot für den ATARI ST war eher gering, aber nicht uninteressant.

Dieser Überblick beschränkt sich auf Neuheiten und auf die Verlage, die ST-Bücher im Programm haben. Viele der großen Namen, auch aus dem Ausland, tauchen mangels Präsenz nicht auf.

Markt & Technik aus München, bisher schon mit cirka 15 ST-Titeln am Markt, wird in einer Reihe "Bookware" Bücher mit Software (oder umgekehrt) anbieten. Schon vorhanden sind "ATARI ST Sound-Enhancer" von F.Mathy, ein Sound- und Musik-Editor sowie "MasterTEXT" von G.Möllmann. Letzteres präsentiert eine Textverarbeitung, die vom Leistungsumfang her 1st_Word entspricht, jedoch schneller sein soll. Beide Bücher werden mit Diskette angeboten und sollen teilweise Source-Code und eine umfangreiche Dokumentation für DM 79,- bieten.

Geplant sind als "Bookware" zusätzlich eine Datenbank und eine Tabellenkalkulation. Mit weiteren Titeln zu GFA-BASIC 3.0 und Omikron 3.0 soll der Bereich der BASIC-Sprachen abgedeckt werden.

Bei **Hüthig** aus Heidelberg war das Ange-

bot ebenfalls etwas “diskettenlastig”. Auf solchen nämlich wird die “Tutor”-Reihe vertrieben, die mit einem “C-Tutor” und einem “Assembler-Tutor” eröffnet wurde. Bei beiden Titeln von Heinrich Kersten handelt es sich jeweils um Einführungs- und Programmierkurse auf Diskette für DM 38,- (als empfohlenem Verkaufspreis, da die Produkte nicht mehr in der Preisbindung für Bücher sind). Für den C-Kurs muß ein Compiler zusätzlich vorhanden sein; dem “Assembler-Tutor” liegt ein Public Domain-Assembler bei. Als dritter Titel der Reihe ist ein Tutor für die GEM-Programmierung unter Omikron-Basic angekündigt.

Bis zur CeBIT werden weitere Buchtitel für den ST erscheinen, darunter das “Programmierlexikon für den ATARI ST” in erweiterter zweiter Auflage. Als einziger Verlag warb **Hüthig** mit einem speziellen Katalog für die ST-Produkte.

Der Hannoversche **Heise-Verlag** hatte zur Buchmesse keine speziellen ST-Titel vorzustellen, allerdings finden sich im Katalog einige Ankündigungen bis zum Frühjahr. Das schon größere Angebot zur Künstlichen Intelligenz wurde um “Gehirn, Sprache und Computer” von Adolf Ebeling (DM 46,-) erweitert, einer kritischen Gegenüberstellung von künstlicher und natürlicher Intelligenz.

SYBEX aus Düsseldorf stellte lediglich die zweite Auflage des “ATARI ST Profibuchs” vor, das inzwischen auf über 1000 Seiten angewachsen ist. Weitere Titel für den ATARI sind nicht geplant; man hält die Angebotspalette für komplett.

Für die meisten Verlage liegt der Saisonschwerpunkt für Neuvorstellungen auf der CeBIT in Hannover. Damit läßt sich erklären, daß das Angebot eher gering war, und daß nicht alle in diesem Bereich tätigen Verlage präsent waren. So war z.B. DATA BECKER im dritten Jahr nicht auf der Buchmesse anwesend.

Insgesamt scheint für die meisten Verlage der ATARI ST “buchmäßig” abgedeckt. Bei vielen Ausstellern war zu hören, daß das Programm komplett sei und man zunächst keine weiteren Titel plane. Nachdem ATARI Omikron.BASIC als Standard vertreibt, werden allerdings noch einige Bücher zu diesem Thema erscheinen.

Eine sehr offene Antwort auf Nachfragen zum ATARI ST hatte der Franzis-Verlag bereit. Sein Vertreter erklärte schlicht, daß sich der Verlag nur noch mit “professionellen” Systemen beschäftigen wolle, zu denen der ST offensichtlich nicht gerechnet wird. Ob sich allerdings jeder der inzwischen 300 000 ST-Anwender nur mit Video-Spielen beschäftigt und dem Trend zum Zweitbuch nicht folgt, mag bezweifelt werden.

Besprechungen der Neuerscheinungen, die für den ST-Benutzer interessant sind, werden Sie in den nächsten Monaten in ST Computer finden.

RT

DIE SYSTEM-VARIABLEN DES TOS

Teil 2

Heute kommen wir zum zweiten und letzten Teil der Besprechung der Systemvariablen.

Resetresidente Programme

Die Systemvariablen unterstützen eine Möglichkeit, wie Programme einen Reset überleben können (Tab. 1). Unmittelbar nach dem Reset werden zuerst alle Interrupts gesperrt und die Peripheriebausteine mit dem *'reset'*-Befehl des 68000 zurückgesetzt. Ein eventuell vorhandenes "Diagnose-ROM-Modul" wird aufgerufen. Falls es sich um einen Warmstart handelt, wird *'memcntrl'* in das MMU-Register übertragen, um die alte Speicherkonfiguration wieder herzustellen.

Wenn *'resvalid'* den Magicwert $31415926 enthält (hier war wohl ein Mathematiker am Werk, denn das sind die ersten Stellen der Zahl Pi), wird noch geprüft, ob der Vektor *'resvector'* eine sinnvolle Adresse enthält (Bits 0 und 24-31 gelöscht). Dann erfolgt der Aufruf der durch ihn bezeichneten Routine.

Dabei muß man beachten, daß ansonsten noch keinerlei Systeminitialisierungen gemacht wurden.

Wenn man nicht gerade das ganze System selbst übernehmen will, möchte man wahrscheinlich nach einigen kleinen Aktionen wieder in die normale Reset-Routine zurückkehren.

Hierzu gibt es zwei Möglichkeiten. Mit *'jmp (A6)'* springt man vor die Abfrage von *'resvalid'*. Das ist nur möglich, wenn man *'resvalid'* vorher ungültig macht, weil man sonst immer wieder von Neuem aufgerufen wird (eine nette Form der Endlosschleife).

Nach einem *'jmp $24(A6)'* dagegen fährt man in der normalen Reset-Routine fort.

Das weicht zwar von der Original-BIOS-Dokumentation ab, doch ist es von ATARI nun selbst so vorgeschlagen worden.

Auch hier ist es natürlich sinnvoll, eine Kette zu bilden: Bei der Installation merkt man sich, ob *'resvalid'* gültig ist, und springt dann am Ende der eigenen Reset-Routine über den gemerkten *'resvector'*.

A6 ist übrigens das einzige Register, das einen definierten Wert hat. Benutzt werden dürfen alle Register. Da noch nicht einmal die Stackzeiger sinnvolle Werte haben, dürfen natürlich keine Unterprogrammaufrufe gemacht werden!

Vertical Blank Interrupt

Der Vertical Blank Interrupt (VBI) wird jedesmal ausgelöst, wenn der Elektronenstrahl des Monitors ein Bild fertig gezeichnet hat, d.h. 50, 60 oder ca. 72 mal pro Sekunde.

Der VBI-Handler des BIOS erledigt hauptsächlich Aufgaben, die die Bilddarstellung beeinflussen. Dazu gehören Cursor-Blinken, der Wechsel der Bildschirmadresse, der Farbpalette oder der Bildschirmauflösung. Ein wenig aus der Rolle fällt nur die Floppy-VBL-Routine.

Würde dies erledigt werden, während das Bild aufgebaut wird, käme es zu unschönen Flackererscheinungen.

Der Interrupt-Handler des BIOS benutzt die Systemvariablen aus (Tab. 2).

In *'_frclock'* wird die Anzahl der seit dem letzten Kaltstart registrierten VBIs mitgezählt.

'_vbclock' dagegen registriert die Zahl der tatsächlich ausgeführten VBI-Routinen. Denn wenn *'vblsem'* auf 0 gesetzt wird, wird der VBI ignoriert (bis auf das Erhöhen von *'_frclock'*). Normalerweise hat *'vblsem'* den Wert 1.

Während der Abarbeitung der VBI-Routine ist *'vblsem'* automatisch auf 0 gesetzt, so daß ein VBI, der vor der Beendigung des alten auftritt, ignoriert wird. Dieser Fall kann aber ohnehin nur dann eintreten, wenn der Interrupt-Level der CPU während einer VBI-Routine heruntergesetzt wird, was normalerweise nicht der Fall ist.

Nachdem alle oben erwähnten Aufgaben des Systems erledigt wurden, werden noch von anderen Programmen gewünschte VBI-Routinen aufgerufen.

Dazu zeigt *'_vblqueue'* auf eine Tabelle mit Adressen von VBI-Routinen. Die Anzahl der Adressen ist durch *'nvbls'* bestimmt. Ein Nullzeiger (0L) in der VBI-Tabelle zeigt an, daß der "VBI-Slot" unbenutzt ist.

Die korrekte Vorgehensweise, um eine eigene VBI-Routine zu installieren, ist wie folgt:

1) die durch *'_vblqueue'* adressierte Tabelle nach einem leeren Slot (Nullzeiger) durchsuchen (*'nvbls'*-Einträge untersuchen).
2) Wenn ein Slot frei ist, einfach gewünschte Adresse eintragen.
3) Wenn alle Slots besetzt sind, gibt es zwei Möglichkeiten. Entweder bricht

das Programm mit einer Fehlermeldung ab, oder es verschiebt die gesamte Tabelle in einen eigenen um einen Eintrag größeren Speicherbereich, trägt sich dort ein und erhöht *'nvbls'* um Eins. Dies ist jedoch nur sinnvoll, wenn das Programm resident im Speicher verbleibt, da ein "Rückverschieben" i.allg. nicht ohne weiteres möglich ist.

4) Um seine eigene Routine wieder zu entfernen, durchsucht man die durch das aktuelle *'_vblqueue'* und *'nvbls'* bestimmte Tabelle nach der Adresse der eigenen Routine und ersetzt sie durch 0L (inzwischen könnte ein anderes Programm ja die Tabelle verschoben haben).

Die VBI-Routinen werden im Supervisor-Modus aufgerufen und dürfen alle Register (außer SSP und USP) verändern.

GEM installiert ebenfalls eine eigene VBI-Routine, die für das Neuzeichnen des Mauszeigers sorgt.

Dabei hält sich GEM jedoch nicht an die Spielregeln. Es benutzt nämlich immer den ersten Slot, auch wenn dieser schon besetzt sein sollte! Daher muß bei AUTO-Ordner-Programmen Punkt 1) dahingehend abgeändert werden, daß erst ab dem zweiten Slot gesucht wird.

GEM sollte einem hier also nicht als Vorbild dienen. Ebenfalls sollte man davon absehen, sich vor eine andere VBI-Routine in den gleichen Slot zu hängen (durch Verkettung), da man so dieser Routine die Möglichkeit nimmt, sich wieder auszuhängen (oder man wird selbst mit abgehängt, das ist dann Künstlerpech).

Die Standard-VBI-Tabelle liegt übrigens auch in den Systemvariablen (*'_vbl_list'*), allerdings sollte man sie nie direkt, sondern nur über *'_vblqueue'* ansprechen.

Nach Aufruf dieser Anwender-VBI-Routinen wird noch die Hardcopy-Routine aufgerufen, falls ALT-HELP gedrückt wurde (s.u.).

Adr.	Name	Gr.	Std.-Wert	Bedeutung
$426	resvalid	L	$31415926	Magic für 'resvector'
$42A	resvector	L		Vektor für resetresidente Programme

Tab. 1: Resetresidente Programme

Adr.	Name	Gr.	Std.-Wert	Bedeutung
$466	_frclock	L		* Zahl der aufgetretenen VBI-Interrupts
$462	_vbclock	L		* Zahl der ausgeführten VBI-Routinen
$452	vblsem	W	1	0 sperrt VBI-Routinen
$454	nvbls	W	8	Zahl der Einträge in 'vblqueue'
$456	_vblqueue	L	$4CE	Zeiger auf Vektortabelle für VBI-Routinen
$4CE	_vbl_list	8L		* Standard-'vblqueue'

Tab. 2: Vertical Blank-Interrupt

Adr.	Name	Gr.	Std.-Wert	Bedeutung
$44E	_v_bs_ad	L		* Zeiger auf logischen Bildschirm
$45A	colorptr	L	0	Zeiger auf neu zu setzende Farbtabelle
$45E	screenpt	L	0	Zeiger auf neu zu setzende Bildschirmadr.
$44C	sshiftmd	B		* Kopie der akt. Bildschirmauflösung
$44A	defshiftmd	B	0	Defaultauflösung bei Wechsel S/W -> Farbe
$46E	swv_vec	L		Vektor für Wechsel S/W <-> Farbe
$448	palmode	W	0	* legt Fernsehnorm fest (unbenutzt!)

Tab. 3: Bildschirm

Buntes oder Schwarzweißes

Auch zur Verwaltung des Monitors und des Bildes auf ihm gibt es Systemvariablen (Tab. 3).

TOS unterscheidet zwischen einem "logischen" und einem "physikalischen" Bildschirm.

Der physikalische Bildschirm ist derjenige, der gerade auf dem Monitor dargestellt wird. Seine Adresse ist demnach direkt in Registern des Video-Chips (Shifters) zu finden.

Auf den logischen Bildschirm beziehen sich dagegen alle gerade stattfindenden Ausgabeoperationen. Dies sind die Textausgaben des VT52-Emulators und die Grafikausgaben der Line A-Routinen. Seine Adresse ist in *'_v_bs_ad'* abgelegt.

Dieses Konzept ermöglicht es, einen Bildschirm unabhängig von dem gerade tatsächlich angezeigten aufzubauen. In vielen Fällen werden beide Adressen natürlich gleich sein, trotzdem sollte man bei eigenen Routinen, die direkt auf den Bildschirmspeicher zugreifen, stets *'_v_bs_ad'* benutzen.

Ganz böse Praxis ist es, direkt auf das vermeintliche Video-RAM zuzugreifen (auch nicht über *'phystop'* minus 32 kB), da es keineswegs immer dort liegen muß.

Die Bildschirmadressen müssen übrigens immer Vielfache von 256 sein, d.h. die untersten 8 Bits sind immer Null.

Um eine neue Farbpalette (16 Words) zu installieren, schreibt man einfach einen Zeiger auf diese nach *'colorptr'*. Bei der nächsten ausgeführten VBI-Routine wird die Farbpalette in die entsprechenden Register des Shifters übertragen. Dann wird *'colorptr'* automatisch gelöscht.

Ähnlich verhält es sich mit *'screenpt'*, nur daß hier die Adressen des logischen und des physikalischen Bildschirms neu gesetzt werden. Allerdings wird *'screenpt'* anschließend nicht gelöscht, so daß dies bei jedem VBI erneut passiert. Abgesehen von einer gewissen Zeitverschwendung führt das zu Kollisionen mit der XBIOS-Funktion *'Setscreen'*. Daher sollte man entweder nur *'screenpt'* oder nur *'Setscreen'* benutzen und im ersten Fall spätestens beim Terminieren des Programms *y* auf 0 setzen.

Die aktuelle Bildschirmauflösung wird in *'sshiftmd'* aufbewahrt. Dabei stehen 0,1 und 2 für die niedrige, mittlere und hohe Auflösung. Daher sollte *'sshiftmd'* nicht verändert werden, ohne auch das entsprechende Register im Shifter neu zu setzen. Hierfür gibt es außerdem die XBIOS-Funktion *'Setscreen'*.

Der VBI-Handler erkennt an einem be-

sonderen Signal des Video-Ports, ob der ATARI Monochrom-Monitor angeschlossen ist.

Beim Wechsel Schwarzweiß- zu Farbmonitor bestimmt *'defshiftmd'*, ob die niedrige oder mittlere Auflösung eingestellt werden soll. Daher sollte hier 0 oder 1 stehen. Nach dem Kaltstart steht hier 0.

Die neue Auflösung (beim Wechsel zum Monochrom-Monitor natürlich die hohe) wird nun nach *'sshiftmd'* und direkt in den Shifter übertragen.

Um ein Programm von den veränderten Verhältnissen zu unterrichten, wird über den Vektor *'swv_vec'* gesprungen.

Normalerweise ist hier die Reset-Routine eingetragen, d.h. es wird ein Warmstart durchgeführt. Dabei bleiben sowohl *'sshiftmd'* als auch *'defshiftmd'* erhalten, so daß nach dem Reset tatsächlich die gewünschte neue Auflösung vorliegt.

Der Reset ist jedoch nur deshalb notwendig, weil das GEM sich beim Systemstart einmal auf eine bestimmte Auflösung "eingestellt" hat und daher neu initialisiert werden muß. Beim Wechsel zwischen den beiden Farbmodi über den Desktop ist dies noch möglich, aber aus der VBI-Routine heraus, wo der Monitorwechsel bemerkt wird, nicht.

Programme, die ganz ohne GEM auskommen (vornehmlich Spiele), können jedoch mit einem Monitorwechsel besser umgehen. Daher sollten Sie in *'swv_vec'* eine eigene Routine eintragen. Hier ist es natürlich nicht sinnvoll, eine Kette zu bilden, daher reicht es, die alte Adresse zu retten und *'swv_vec'* beim Verlassen des Programms wieder zu restaurieren. Jetzt wird es kritisch, weil GEM nicht mit der veränderten Auflösung zurechtkommt.

Nach Abarbeitung der *'swv_vec'*-Routine wird ganz normal im VBI-Handler fortgefahren.

Die Routine wird im Supervisor-Mode aufgerufen und darf alle Register (außer SSP und USP) verändern.

In *'palmode'* schließlich soll die Fernsehnorm festgelegt sein, mit der ein eventuell angeschlossener HF-Modulator arbeitet. 0 steht für die NTSC-Norm, alles andere bedeutet PAL-Norm. Soweit mir bekannt, steht hier immer 0. Ein ähnliches Flag findet sich stattdessen im "system header block" (Abb. 2 des 1. Teils).

Konsole - Zeichen für Zeichen

Die "Konsole" des ST, bestehend aus Tastatur und dem über den VT52-Emulator verwalteten Bildschirm, hat auch ihr Plätzchen im Variablendschungel gefunden (Tab. 4).

'conterm' ist ein Bitvektor, der einige Optionen steuert. Der Standardwert ist 7.

Bei gesetztem Bit 0 ertönt bei der Ausgabe von CTRL-G die "Glocke". Der GEM-Anwender hört diesen Sound, wenn er außerhalb einer Dialog-Box herumklickt. Sonst bleibt CTRL-G stumm.

Die automatische Tastenwiederholung läßt sich durch ein gesetztes Bit 1 aktivieren.

Der Tastenklick ertönt nur bei gesetztem Bit 2.

Normalerweise geben BIOS und GEMDOS bei der Eingabe von der Tastatur nur ASCII- und Scan-Kode zurück. Bei gesetztem Bit 3 wird zusätzlich in den Bits 31-24 der Rückgabewerte der aktuelle Tastenstatus wie ihn auch *'Kbshift'* liefert, abgelegt.

Über den Vektor *'con_state'* springt das BIOS zu Beginn der *'Bconout'*-Routine für die Konsole. In den Bits 7-0 von D1 findet man das auszugebende Zeichen, die Bits 31-8 sind immer Null (die Bits 15-8 des an *'Bconout'* übergebenen Zeichens gehen verloren).

Die normale Routine gibt die druckbaren Zeichen aus und wertet alle CTRL-Codes und ESC-Sequenzen aus. Nach Registrierung eines ESC setzt sie einfach den *'con_state'*-Vektor auf eine spezielle ESC-Behandlungsroutine um. Dadurch wird das nachfolgende Zeichen als zum ESC gehörig erkannt und ausgewertet, so daß das *'A'* aus *'ESC A'* nicht als *'A'* auf dem Bildschirm erscheint.

Alle ESC-Sequenzen außer ESC Y bestehen nur aus einem weiteren Zeichen. Daher wird *'con_state'* wieder auf die normale Routine zurückgesetzt und die Funktion ausgeführt.

Bei ESC Y (Setzen der Cursor-Position) wird *'con_state'* auf eine weitere Routine gesetzt, die nichts weiter macht, als das nächste Zeichen, die neue Cursor-Zeile, in *'save_row'* zwischenzuspeichern und *'con_state'* erneut umzusetzen. Erst beim nächsten Zeichen, der neuen Cursor-Spalte, ist die Sequenz beendet, *'con_state'* wird wieder auf den Ausgangswert gesetzt und die Funktion ausgeführt.

Man kann hier eine eigene Routine einhängen, die aber dann zumindest kein ESC mehr an die Original-Routine "durchlassen" darf, da diese dann sofort den *'con_state'*-Vektor umsetzt, und die eigene Routine damit nicht mehr zum Zuge kommt.

Die *'con_state'*-Routine wird im Supervisor-Modus aufgerufen und darf alle Register (außer SSP und USP) verändern.

Hardcopy oder 30 Sekunden Pause?

Tab. 5 gibt Aufschluß über die mit der Bildschirm-Hardcopy verbundenen Systemvariablen.

Der Zähler *'_dumpflg'* hält den Zustand der ALT-HELP-Tastendrücke bzw. der Hardcopy-Routine fest.

Adr. Name	Gr.	Std.-Wert	Bedeutung
$484 conterm	B	7	Bitvektor für Funktionen der Konsole
$4A8 constate	L		Vektor für Ausgabe auf Konsole
$4AC sav_row	W		* Zwischenspeicher für Cursorzeile bei ESC Y

Tab. 4: Konsole

Adr. Name	Gr.	Std.-Wert	Bedeutung
$4EE _dumpflg	W	-1	Status der Hardcopy-Routine
$502 scr_dump	L		Vektor für Hardcopy ('Scrdmp')
$506 prv_lsto	L		Vektor für Ausgabestatus paralleler Port
$50A prv_lst	L		Vektor für Ausgabe auf parallelem Port
$50E prv_auxo	L		Vektor für Ausgabestatus serieller Port
$512 prv_aux	L		Vektor für Ausgabe auf seriellem Port

Tab. 5: Hardcopy

Er steht auf -1 im "Normalzustand" und wird von der Interrupt-Routine, die Tastendrücke verarbeitet, um Eins höher gezählt, jedesmal wenn ALT-HELP gedrückt wird.

Demnach bedeutet eine 0, daß mit dem Ausdruck der Hardcopy begonnen werden soll.

Am Ende des VBI-Handlers wird, wenn *'_dumpflg'* 0 ist, die Hardcopy-Routine *'Scrdmp'* (XBIOS #20) aufgerufen. *'Scrdmp'* ruft die eigentliche Hardcopy-Routine *'Prtblk'* (XBIOS #36) auf.

Vorher stellt sie noch den von *'Prtblk'* erwarteten Parameterblock zusammen und setzt *'_dumpflg'* auf 1. Dies signalisiert somit eine laufende Hardcopy.

'Prtblk' bricht die Hardcopy ab, wenn *'_dumpflg'* ungleich 1 wird, also bei einem erneuten ALT-HELP (wobei *'_dumpflg'* ja 2 wird).

Am Ende von *'Prtblk'* wird *'_dumpflg'* auf jeden Fall wieder auf -1 gesetzt.

Daraus folgt, daß man vor dem Aufruf von *'Prtblk'* über das XBIOS *'_dumpflg'* selbst auf 1 setzen muß.

'Scrdmp' springt gleich zu Anfang über den Vektor *'scr_dump'*. Hier kann also eine eigene Hardcopy-Routine installiert werden.

Sie kann entweder ebenfalls *'Prtblk'* aufrufen (mit einem anderen Parameterblock) oder ganz neugeschrieben sein. Sie muß noch nicht einmal *'_dumpflg'* zurücksetzen, da dies von *'Scrdmp'* nachher schon erledigt wird.

Die *'scr_dump'*-Routine wird im Supervisor-Modus aufgerufen und darf die Register D0-D2 und A0-A2 verändern.

'Prtblk' benutzt zur Ausgabe der einzelnen Zeichen an den Drucker nicht direkt die BIOS-Funktion *'Bconout'*, sondern hat eigene Ausgabe-Routinen, die über die Vektoren *'prv_lst'* oder *'prv_aux'* springen, je nachdem ob der Drucker parallel oder seriell angeschlossen ist (dies kann im Parameterblock von *'Prtblk'* festgelegt werden).

Normalerweise sind hier die Adressen der normalen Ausgabe-Routinen eingetragen, die auch *'Bconout'* aufruft. Bei *'prv_aux'* ist dies allerdings ein Fehler, da hier noch die Abfrage der BIOS-Konfiguration (einstellbar etwa mit dem Kontrollfeld-Accessory bzw. der XBIOS-Funktion *'Setprt'*) erfolgt. Somit ist eine Ausgabe über den seriellen Port nur möglich, wenn der serielle Drucker sowohl mit *'Setprt'* als auch im Parameterblock angewählt wurde.

Sinnvoll wäre es, wenn alle Programme ihre Grafikausgaben an Drucker über diese Routinen leiten würden. Dann wären Drucker-treiber, die sich in *'Bconout'* einhängen und bestimmte Code-Umwandlungen vornehmen, des Problems entledigt, daß solche Umwandlungen bei Grafikdaten nicht erwünscht sind.

Ferner hat die Hardcopy-Funktion auch ihre eigenen Routinen für den Ausgabestatus (*'prv_lsto'* bzw. *'prv_auxo'*). Hier stehen natürlich normalerweise die entsprechenden Routinen, die auch *'Bcostat'* aufruft. Diese Routinen werden jedoch von *'Prtblk'* nie aufgerufen! Das lästige Warten nach einem ALT-HELP, wenn der Drucker nicht eingeschaltet ist, könnte z.B. vermieden werden, wenn *'Prtblk'* gleich zu Beginn mit diesen Funktionen feststellen würde, ob der Drucker überhaupt annahmebereit ist, und gegebenenfalls abbrechen würde.

Dies wäre schon eine Anwendung für eine eigene *'scr_dump'*-Routine, die genau diese Abfrage durchführt, bevor sie in die normale *'scr_dump'*-Routine zurückkehrt.

Eigene Hardcopy-Ausgabe-Routinen müssen sich einfach wie die entsprechenden BIOS-Funktionen verhalten, d.h. das auszugebende Zeichen steht ab 6(SP). Die normalerweise bei 4(SP) stehende Gerätenummer ist hier undefiniert.

Für eigene Hardcopy-Ausgabestatus-Routinen gilt Entsprechendes.

Ereignisvektoren

Es gibt einige Routinen, die beim Eintreten spezieller Ereignisse (*"events"*) aufgerufen werden (Tab. 6).

Der *'etv_timer'* wird gleich im Zusammenhang mit dem 200-Hz-Interrupt erklärt.

Bei Fehlern während Diskettenoperationen ruft das BIOS den "Critical Error Handler" (CEH) auf, der in *'etv_critic'* festgelegt ist. Er soll "kritische" Fehler handhaben und entscheiden, wie darauf reagiert werden soll.

Das TOS hat einen CEH für TOS und einen für GEM-Programme. Der letztere, vom AES installiert, gibt die bekannten Alert-Boxen wie *"Floppy A: antwortet nicht..."* aus.

Eigene Laufwerkstreiber oder Programme können den CEH auch ohne weiteres aufrufen.

Auf dem Stack werden dem CEH die BIOS-Fehlernummer (-1...-31) und die Laufwerkskennung (0 für A: usw.) übergeben (beides 'words').

Außerdem kann im Register D0 eine Default-Fehlernummer übergeben werden. Dies wird vom CEH des AES aber ignoriert.

Der CEH muß im Supervisor-Modus aufgerufen werden und darf die Register A0-A2 und D0-D2 verändern.

Er liefert ein *'longword'* zurück. Dies kann eine BIOS-Fehlernummer sein, die der Aufrufer seinerseits zurückgeben soll. Auch 0 ist erlaubt, um zu signalisieren, daß der Fehler ignoriert werden soll. Der AES-CEH gibt hier den ihm übergebenen Fehlercode zurück, wenn der Anwender die Alert-Box mit *'ABBRUCH'* beendet.

Ein Wert von $10000 zeigt an, daß die fehlerhafte Operation noch einmal wiederholt werden soll. Der CEH des AES liefert dies, wenn *'WEITER'* ausgewählt wurde.

Ein eigener CEH kann auch installiert werden,ist praktikabel aber nur für TOS-Programme. Dazu setzt ein residentes AUTO-Ordner-Programm *'etv_critic'* einfach um. Der eigene CEH wird aber nur bei TOS-Programmen aktiviert.

Der CEH darf BIOS/XBIOS-, aber keine GEMDOS- oder gar AES-Aufrufe machen.

Bei der Terminierung eines Programms mit der GEMDOS-Funktion *'Pterm'*/

Adr.	Name	Gr.	Std.-Wert		Bedeutung
$400	etv_timer	L			Anwendervektor für 50-Hz-System-Timer
$404	etv_critic	L			Vektor für Behandlung kritischer I/O-Fehler
$408	etv_term	L			Vektor für Prozeßterminierung
$40C	etv_xtra	5L	0	*	reserviert für zukünftige 'etv'-Vektoren

Tab. 6: Ereignisvektoren

Adr.	Name	Gr.	Std.-Wert	Bedeutung
$4BA	_hz_200	L		* Zahl der 200-Hz-Interrupts
$442	_timer_ms	W	20	* ms zwischen zwei Systemtimer-Interrupts

Tab. 7: 200-Hz-Systeminterrupt

Adr.	Name	Gr.	Std.-Wert	Bedeutung
$4A2	savptr	L		Zeiger auf BIOS-Stack für Register
$4B2	_bufl	2 L	2	Zeiger auf GEMDOS-BCB-Listen
$516	pun_ptr	L		Zeiger auf Harddisk-Information
$51E	bis_vec	8 L		Vektoren für 'Bconstat'
$53E	bi_vec	8 L		Vektoren für 'Bconin'
$55E	bos_vec	8 L		Vektoren für 'Bcostat'
$57E	bo_vec	8 L		Vektoren für 'Bconout'

Tab. 8: Sonstiges

'Pterm0'/'Ptermres' wird eine in *'etv_term'* installierte Routine aufgerufen.

Sie kann vom Programm zur Verfügung gestellt werden, um eine korrekte Terminierung auch in unerwarteten Situationen (CTRL-C oder Bomben) zu gewährleisten. Dazu gehört z.B. das Restaurieren veränderter Sprungvektoren.

'etv_term' wird im Supervisor-Modus aufgerufen, darf die Register D0-D2 und A0-A2 verändern und endet normalerweise mit *'RTS'*. Vorher sollte sie sich selbstverständlich deinstalliert haben.

Es ist allerdings auch möglich, direkt ins Programm zurückzuspringen, um das *'Pterm'* zu ignorieren. Dazu müssen natürlich alle Register usw. restauriert werden (in C macht man das mit einem *'longjmp'*).

Die *'etv_term'*-Routine darf GEMDOS-Aufrufe nur machen, wenn sie nicht mehr zurückkehrt. BIOS/XBIOS-Aufrufe sind dagegen ohne weiteres möglich.

Bei *'Pterm'/'Pterm0'* wird *'etv_term'* vor allem anderen aufgerufen, bei *'Ptermres'* sind allerdings schon die dem Prozeß gehörenden Speicherbereiche vor der Freigabe geschützt worden (die Memory-Descriptoren wurden aus der Belegt-Liste entfernt).

Normalerweise zeigt *'etv_term'* einfach nur auf *'RTS'*.

Ab *'etv_xtra'* ist Platz für 5 weitere, zukünftige Event-Vektoren reserviert.

200-Hz-Interrupt

TOS erledigt noch einige weitere periodische Aufgaben, die unabhängig von der Bildwiederholfrequenz laufen oder eine höhere Priorität als der VBI haben sollen.

Daher ist der Timer C im MFP so programmiert, daß er alle 5 Millisekunden (also 200 mal pro Sekunde) einen Interrupt auslöst.

Die Variablen hierzu sind in Tab. 7 aufgeführt.

Die Anzahl dieser Interrupts seit dem letzten Kaltstart wird in *'_hz_200'* mitgezählt. Dies ist also das Analogon zu *'_frclock'*, aber wesentlich besser geeignet, um Zeitmessungen vorzunehmen, da die Frequenz hier immer die gleiche ist. Die Zeitdauer (eines Benchmarks z.B.) erhält man also durch einfache Differenzbildung der *'_hz_200'*-Werte zu Beginn und zum Ende.

Um Überlaufprobleme braucht man sich nicht zu kümmern, da *'_hz_200'* erst nach 248,6 Tagen einmal durchgezählt ist!

'_hz_200' wird vom BIOS selbst für verschiedene Zeitmessungen benutzt.

Direkt mit 200 Hz werden sonst keine periodischen Aufgaben durchgeführt. Bei jedem vierten 200 Hz-Interrupt, also mit 50 Hz, wird aber noch einiges erledigt. Dazu gehören die Soundverarbeitung (im Hintergrund ablaufende "Musik" einschließlich des Tastaturklicks) und die automatische Tastenwiederholung (hier wird das Bit 1 von *'conterm'* abgefragt).

Zum Schluß werden noch vom Anwender definierbare Routinen aufgerufen. Dazu wird über den *'etv_timer'*-Vektor gesprungen. Hier kann man sich also - in einer Kette - einhängen. Als erstes Argument auf dem Stack (Word bei 4(SP)) bekommt man die Zahl der Millisekunden seit dem letzten Interrupt übergeben. Tatsächlich handelt es sich dabei um *'_timer_ms'*, das dementsprechend immer 20 ist.

Die Anwender-Routinen werden im Supervisor-Modus aufgerufen und dürfen alle Register (außer den Stackzeigern) verändern.

Beim Systemstart hängen sich sowohl GEMDOS als auch das AES in den *'etv_timer'*-Vektor ein. GEMDOS benutzt dies, um seine Uhr weiterzusetzen (in 2-Sekunden-Schritten).

Dies ist die Uhr, die für die Erstellungszeit bei Dateien verantwortlich ist.

Sonstiges

Die bisher noch nicht besprochenen Systemvariablen sehen Sie in Tab. 8.

Das BIOS ist "reentrant", d.h. während der Abarbeitung eines TRAPs (13 oder 14) darf ein erneuter TRAP ausgelöst werden (von einer Interrupt-Routine oder einem Unterprogramm des BIOS), ohne daß es zu Konflikten beim Retten oder Restaurieren der Arbeitsumgebung der CPU kommt.

Das heißt natürlich nicht, daß z.B. mitten in einem Floppy-Zugriff plötzlich eine weitere Floppy-Operation gestartet werden darf. Auf solche Kollisionen muß man selbst achten, so daß man sich meistens auf "harmlose" BIOS-Operationen beschränkt (z.B. *'Kbshift'*, *'Physbase'* usw.).

Es sei noch angemerkt, daß GEMDOS und AES in keiner Weise reentrant sind, so daß ein GEMDOS-Aufruf aus einer Interrupt-Routine mehr einem Lottospiel gleicht, wobei die Gewinnchancen allerdings bedeutend höher sind.

Bei einem TRAP befinden sich auf dem Stack bekanntlich die Parameter für den jeweiligen Funktionsaufruf, daher können die zu rettenden Register nicht so ohne weiteres ebenfalls dort abgelegt werden, da sich dann die Offsets für die Parameter verschieben würden.

BIOS hat extra hierfür einen eigenen kleinen Stack. *'savptr'* zeigt auf dessen Spitze. Bei einem TRAP 13 oder 14 werden nun einige Register (SR,PC,D3-D7,A3-A7) ab *'savptr'* abwärts dort abgelegt.

Danach ist *'savptr'* also um 46 erniedrigt worden. Am Ende des TRAPs werden dementsprechend alle Register restauriert und *'savptr'* wird wieder erhöht.

Die maximal erlaubte Schachtelungstiefe von TRAPs liegt bei 6 (die Original-ATARI-Dokumentation spricht von 3), dies dürfte wohl mehr als genug sein.

Allerdings ist der TRAP-Aufruf von Interrupt-Routinen aus doch nicht so ohne weiteres möglich, da ein Interrupt auch während des Rettens der Register auftreten kann. Da *'savptr'* noch nicht erniedrigt wurde, werden schon gerettete Register bei einem nachfolgenden TRAP überschrieben. Hier bräuchten bloß während des Arbeitens mit dem *'savptr'* alle Interrupts gesperrt zu werden.

Dieser Fehler kann umgangen werden, indem man sich in Interrupt-Routinen den *'savptr'* merkt, auf einen eigenen kleinen Speicherbereich (46 oder sicherheitshalber 92 Byte groß) umsetzt, seine BIOS-Aufrufe macht und anschließend *'savptr'* wieder restauriert.

Seit einiger Zeit macht der ATARI-Harddisktreiber eine Struktur über *'pun_ptr'* verfügbar.

Zuerst kommt ein *'word'*, das die Anzahl der gefundenen physikalischen Harddisks angibt. Danach folgen 16 Bytes (für die Laufwerke A: bis P:), die angeben, ob das Laufwerk vom Harddisk-treiber verwaltet wird (gleich 0) oder nicht (ungleich 0).

Ohne ATARI-Harddisktreiber ist *'pun_ptr'* wie zu erwarten ein Nullzeiger.

'_bufl' sind zwei Zeiger auf sogenannte BCB-Listen. Dies sind Strukturen, mit denen GEMDOS gepufferte Sektoren verwaltet. Dieses Thema ist zu umfangreich, um hier behandelt zu werden. Ich verweise daher auf [1].

Adr.	Name	Gr.	Std.-Wert	Bedeutung
$486	trp14ret	L	0	* Return-Adresse für TRAP14-Handler ?
$48A	criticret	L	0	* Return-Adresse für CEH ?
$49E	_____md	L		* Platz für mehr MDs
$4AE	sav_context	L	0	* Beginn Post-Mortem-Bereich
$4BE	the_env	L	0	* Zeiger auf Default-environment
$4CA	_auto_path	L	0	* Zeiger auf Zugriffspfad für AUTO-Ordner
$4F0	prt_abt	W	0	* Abbruch-Flag für Hardcopy ?
$4F6	_shell_p	L	0	* Zeiger auf Arbeitsumgebung r Shellde

Tab. 9: Nie Benutztes

Seit Blitter-TOS gibt es eine einfache Möglichkeit, die BIOS-Funktionen für die zeichenrientierten Geräte (*'Bconin'*, *'Bconout'*, *'Bconstat'* und *'Bcostat'*) zu modifizieren bzw. durch eigene zu ersetzen.

Es gibt hierzu 4 Vektortabellen mit den Adressen der jeweiligen Routinen der BIOS-Geräte 0 bis 7.

So ist *bo_vec[2]* z.B. der Vektor für die Ausgabe eines Zeichens auf dem Bildschirm.

BIOS unterstützt eigentlich nur die Gerätenummern 0 bis 5 und auch die nicht bei allen Funktionen. Nicht definierte Unterfunktionen und die Routinen für die Geräte 6 und 7 bestehen nur aus einem *'RTS'* und geben daher undefinierte Werte zurück.

Die Routinen werden im Supervisor-Modus aufgerufen und die Register D0-D2 und A0-A2 dürfen verändert werden. Die Parameter beginnen bei 4(SP) mit der Gerätenummer, die hier nicht mehr beachtet werden muß.

Unbenutzte Systemvariablen...

...gibt es nämlich auch. Einige haben wir schon kennengelernt, eine komplette Übersicht bietet (Tab. 9). Da sie von ATARI alle benannt wurden, kann man spekulieren, ob sie nur geplant oder nicht mehr notwendig sind.

Ferner ist der Bereich von $3EC bis $3FF noch nicht genutzt.

Das Ende der benutzten Systemvariablen liegt zur Zeit bei $59E, was natürlich nicht so bleiben muß.

Abschließend möchte ich noch hinzufügen, daß mir die Auswahl und Zusammenstellung der Systemvariablen recht wahllos und unüberlegt zu sein scheint.

Bei einigen scheint die Nützlichkeit äußerst fragwürdig zu sein (wie z.B. *'themd'*), bei anderen ist die Beeinflußbarkeit des TOS geringer als erwartet (z.B. *'seekrate'*).

An andere wichtige Variablen, die man oft vermißt, kommt man meist legal überhaupt nicht heran. Hier seien als Beispiel nur der Schreibschutz-Status sowie der auf Grund dessen ermittelte Media-change-Status der Floppies erwähnt.

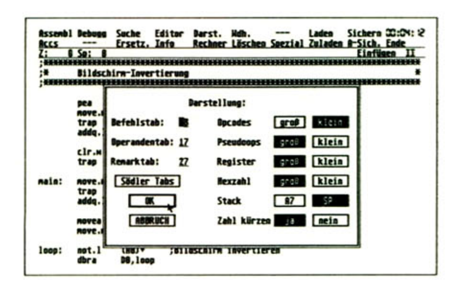
Alex Esser

Literatur:

[1] Auf der Schwelle zum Licht, Teil 3, ST-Computer 3/88

ENDE

Programmierfreude Byte für Byte.



Der neue Assembler: Für alle Programmierer, denen die Zeit im Nacken sitzt, gibt es jetzt einen Assembler, der nicht nur fixe Programme erzeugt, sondern auch durch seine eigene Geschwindigkeit verblüfft: Der OMIKRON. Assembler übersetzt durchschnittlich 1,1 Millionen Zeilen pro Minute und ist damit der Schnellste auf dem ATARI ST.

Schnelle Assemblierung allein nutzt aber wenig, wenn die gewonnene Zeit beim Editieren und Debuggen wieder verlorengeht. Deshalb präsentiert sich der OMIKRON. Assembler als ein Entwicklungssystem, das den Programmierer in jeder Phase optimal unterstützt.

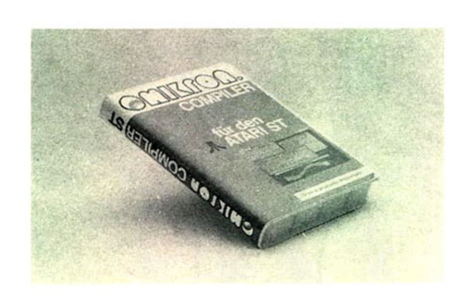
Der integrierte Editor sorgt für fixe Eingabe: auf Tippfehler oder falsche Adressierung wird sofort hingewiesen, die Formatierung erfolgt selbsttätig.

Der Assembler verdaut Labelnamen mit bis zu 23 Zeichen, unterscheidet nicht nur Groß- und Kleinschreibung, sondern kennt auch Umlaute und sorgt so für lesbare Listings. Falls beim Assemblieren Fehler auftauchen, merkt er sich die entsprechenden Stellen. Im Editor bringt Sie dann ein Tastendruck in die betreffenden Zeilen.

Besondere Erwähnung verdient der symbolische Debugger: auch wenn Ihr Programm das Betriebssystem völlig durcheinandergebracht hat, arbeitet er weiter. Komfortabel mit Mausbedienung, Screen-Editing und List-Scroll. UNTRACE mit Abbruchbedingung kreist Fehler systematisch ein: die letzten 256 ausgeführten Befehle lassen sich mitsamt den jeweiligen Registerinhalten auflisten.

Und das gab's noch nie: mit einem Tastendruck springen Sie jederzeit aus dem Debugger an die dem PC entsprechende Stelle im Quelltext. In Verbindung mit den kurzen Assemblierzeiten steht so Korrekturen direkt im Quelltext nichts mehr im Wege.

OMIKRON. Assembler, DM 99,–
OMIKRON. Assembler (Makroversion, lieferbar Januar '89,
Update gegen Differenzbetrag), DM 198,–



Der bewährte Compiler: Wenn Sie mehr Tempo in Ihre Programme bringen wollen, dürfte Sie der OMIKRON.BASIC Compiler interessieren. Er übersetzt ein BASIC-Programm in kompakten Maschinencode, der besonders flott ausgeführt wird: oft schneller als ein vergleichbares C-Programm. Gleichzeitig werden Ihre BASIC-Programme selbständig lauffähig.

Besonders groß ist der Geschwindigkeitszuwachs, wenn Ihr Programm viele Integer-Variablen verwendet. Mit einer Zunahme um den Faktor 10 bis 20 können Sie dann rechnen.

Falls Sie mehr mit Fließkomma-Zahlen arbeiten, möchten wir Ihnen den OMIKRON. BASIC Compiler 68881 nahelegen, der sich bestens mit dem mathematischen Co-Prozessor MC68881 versteht. Ihre BASIC-Programme sprechen diesen Prozessor direkt an, Fließkomma-Berechnungen werden enorm beschleunigt. Alle Funktionen des normalen Compilers sind natürlich enthalten.

OMIKRON.BASIC Compiler, DM 179,–
OMIKRON.BASIC Compiler 68881, DM 229,–

OMIKRON.BASIC (Modul) DM 229,– ·
MIDI-, Statistik-, Numerik-, Complex-, EasyGEM-Library je DM 79,– · Finanzmathematik-, Inline-Assembler-Library je DM 129,– ·
Maskeneditor, Link_it, D.R.A.W.! je DM 79,– ·
OMIKRON.Software, Erlachstraße 15a, D-7534 Birkenfeld 2
XEST, Webgasse 21, A-1060 Wien
Thali AG, Industriestrasse 6, CH-6285 Hitzkirch



Das offizielle ATARI-BASIC.

Flexible Modulprogrammierung mit ADIMENS Talk

Teil 4

SUCHEN VON DATENSÄTZEN

Flexible Modulprogrammierung mit ADIMENS Talk Teil 4 Suchen von Datensätzen Willkommen zum 4. Teil der flexiblen Modulprogrammierung mit ADIMENS Talk. Nachdem Sie bereits Kenntnisse über die Möglichkeiten des Datenbanksystems besitzen und die flexiblen Programmoduln als ein Instrument zur schnellen Entwicklung von EDV-Projektenzu schätzen wissen, wird Ihnen in diesem Teil ein weiteres, sehr reizvolles Programmodul vorgestellt.

Beim gründlichen Studium dieser Serie werden Sie sicher festgestellt haben, daß die flexiblen Programmoduln eine Schlüsselfunktion beim Einsatz Ihres Datenbanksystems bilden. Denn sie lassen sich gemäß dem TOP-DOWN-Entwurf immer wieder in Ihrer Grundstruktur verwenden.

Die gespeicherten Informationen bilden im allgemeinen die Grundlage für die vielfältigsten Anwendungen. Unabhängig davon, ob Sie Adressen für eine Mail-Merging-Aktion benötigen, Umsatzberichte zu erstellen haben, ob Laborwerte zur Analyse anstehen oder Stücklisten in der Fertigung zusammengestellt werden müssen, immer erfolgt der Zugriff auf eine Datenbank.

Nicht einfach ist es, aus den vielfältigen Möglichkeiten für das interaktiv gesteuerte Suchen dieser unterschiedlichen Daten verschiedener Anwendungsgebiete ein geeignetes Lösungskonzept zu erstellen. In diesem Teil wird deshalb ein Suchmodul vorgestellt werden, das Ihnen erlaubt, nach zwei Merkmalen zu suchen, die sowohl numerischer als auch alphanumerischer Form sein können.

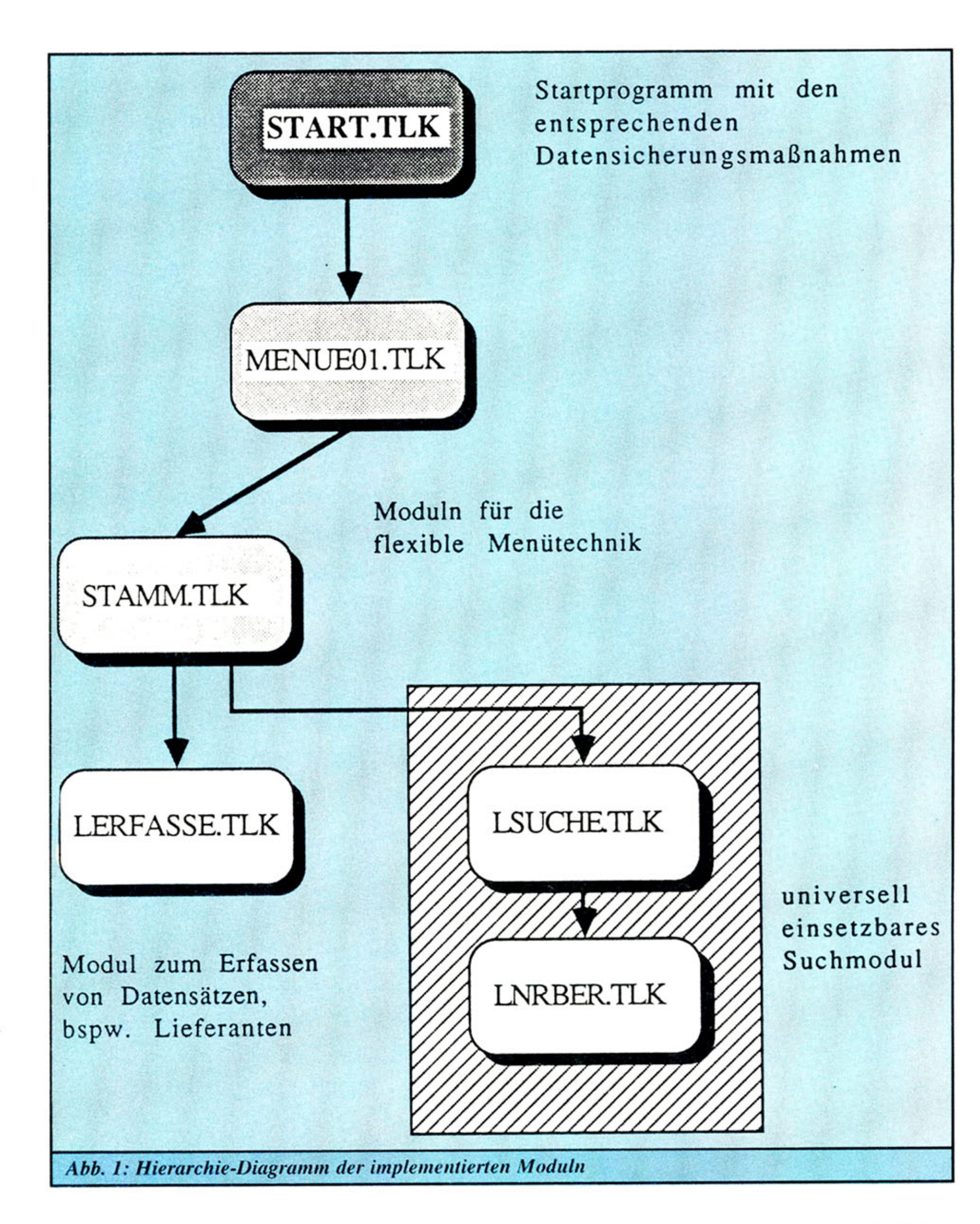


Abb. 1: Hierarchie-Diagramm der implementierten Moduln

In den letzten Teilen wurden die Moduln START.TLK, MENUE01.TLK, STAMM.TLK, und LERFASSE.TLK bereits vorgestellt. Das Suchmodul ist ein Beispiel für ein Modul zum Suchen erfaßter Datensätze und wird als LSUCHE. TLK bezeichnet. Das Programm zur Plausibilitätskontrolle der fragmentarischen Eingabe einer Nummer wird als LNRBER.TLK bezeichnet. Damit der Überblick über die Hierarchie der Moduln nicht verlorengeht, habe ich Ihnen in Abb. 1 das Hierarchie-Diagramm der bisher implementierten Moduln zusammengestellt.

Vorstellung des Modul-Konzeptes

Die flexible, universell einsetzbare Suche nach Datensätzen wurde so geplant, daß die Benutzerfreundlichkeit gewährleistet ist, die aber trotzdem noch alle Möglichkeiten der differenzierten Selektion von bestimmten Datensätzen bestehen. Deshalb hat man hier die Möglichkeit, nach einem numerischen wie der Liefernummer und einem alphanumerischen Merkmal wie dem Namen des Lieferanten zu suchen. Doch nicht genug der Vielfalt. Dieses Modul erlaubt auch noch die differenzierte Auswahl von Lieferantennummern. Diese Lieferantennummern werden aufsteigend immer um Eins erhöht. Wenn z.B. ein Lieferant ein bestimmtes Produkt nicht mehr liefert, wird der Datensatz gelöscht. Die Anzahl der Datensätze fällt deshalb nicht immer mit der letzten Liefernummer zusammen.

Man hat nun mehrere Möglichkeiten, nach den unterschiedlichsten Kriterien die Daten zu suchen.

- Es kann nur nach einer bestimmten Lieferantennummer gesucht werden.
- Es kann eine fragmentarische Lieferantennummer angegeben werden.
- Es kann eine differenzierte Bereichsauswahl der Liefernummer getroffen werden.
- Es können fragmentarische Eingaben gemacht werden. d.h. es können z.B. alle Lieferanten gesucht werden, welche mit "M" anfangen.

Dazu können die Eingaben selbstverständlich untereinander kombiniert werden. Der Leser bemerkt an dieser Stelle sicher, daß er hier ein sehr komplexes Werkzeug zur Datensatzsuche zur Hand haben wird.

An dieser Stelle soll allerdings erwähnt werden, daß eine geeignte und gut struk-

```
================================================================================
Datum                  DATEN - SUCHE                *** V 1.01 ***       Uhrzeit
03.07.88      flexible Anwendungsprogrammierung   mit ADIMENS Talk      18:40:58
--------------------------------------------------------------------------------
                     SUCHEN  neuer Lieferanten ADRESSEN
================================================================================

 Bitte tragen Sie die Lieferanten Nummer ein :        Letzte Lieferantennr.:13
                     _____
 Bitte tragen Sie den Namen des Lieferanten ein :  Anzahl Datensätze     :13
          _______________

================================================================================
                  Bedeutungen bei fragmentalem Suchen
           '*' z.Bsp 2*** Suche im Bereich von 200 bis 299
           Bei der Nummer werden die Leerzeichen ignoriert.
           '?' z.Bsp. May? Suche Namen, die mit May beginnen
================================================================================
Lieferanten Nr.= 0--> ENDE; Lieferanten Nr.= D--> DIFFERENZIERTE BEREICHSEINGABE

1 INFO 2      3      4      5      6      7      8      9      10 STOP
```

Abb. 2: Bildschirmausdruck nach Auswahl Suchen von Lieferanten-Datensätzen

```
================================================================================
Datum                  DATEN - SUCHE                *** V 1.01 ***       Uhrzeit
03.07.88      flexible Anwendungsprogrammierung   mit ADIMENS Talk      18:49:32
--------------------------------------------------------------------------------
                     SUCHEN  neuer Lieferanten ADRESSEN
================================================================================

 Bitte geben Sie einen Zahlenbereich an.               Letzte Lieferantennr.:13
                                                       Akt. Lieferantennr.  :7
 Unterer Zahlenwert :        4                         Anzahl Datensätze    :13
 Oberer Zahlenwert :         7

================================================================================
 SUCHE im Bereich von : 4 bis 7                         BITTE WARTEN ...........
================================================================================
                    Es wurden 4 Datensätze gefunden
================================================================================
BILDSCHIRM Adressen <F1>  DATENSÄTZE alle <F5>     DRUCKER <F7>        <F10>
 AUSDRUCK  Telefon  <F2>    DRUCKEN  best.<F6>   INITIALISIEREN        ENDE
```

Abb. 3: Bildschirmausdruck bei Suchende

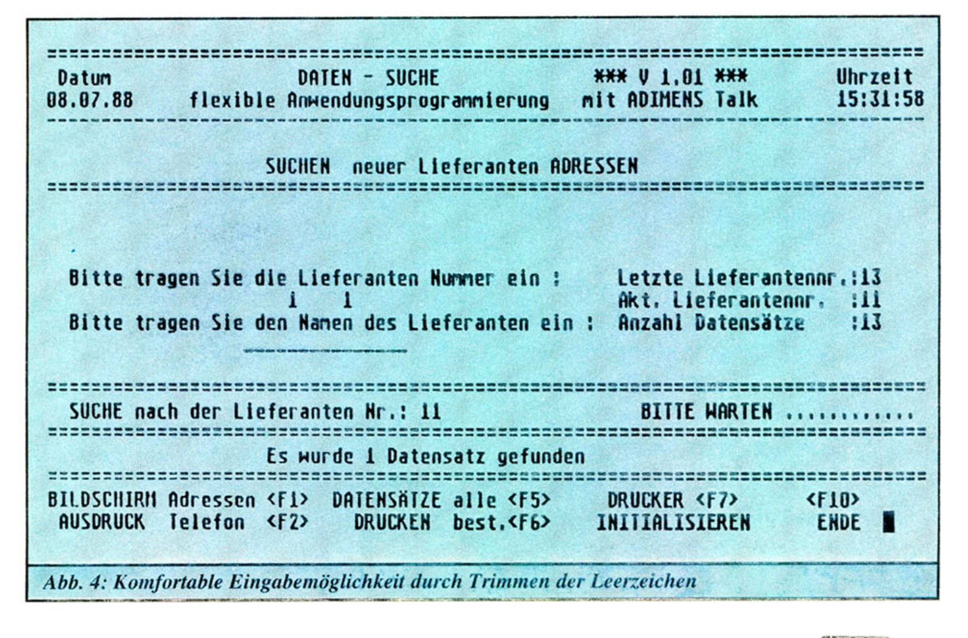


Abb. 4: Komfortable Eingabemöglichkeit durch Trimmen der Leerzeichen

turierte Bildschirmmaske die Arbeitsmotivation erhöht. Wollen Sie Ihren eigenen Entwurf ins Programm umsetzen, sollten Sie die Erkenntnisse der Softwareergonomie beachten. Wenn eine Bildschirmmaske mit Informationen überfrachtet ist, können wichtige Daten untergehen, und der Streß am Bildschirm wird erhöht. Bei der Erstellung einer Maske soll die Benutzerfreundlichkeit und Akzeptanz für den potentiellen Benutzer so groß wie möglich sein. Zu den psychischen Belastungsmomenten zählen alle in irgendeiner Form streßauslösenden oder -verstärkenden Faktoren der Bildschirmarbeit. Dabei spielt Unterforderung durch undifferenzierte und übertriebene Hilfestellungen ebenso eine Rolle wie das Gefühl mentaler Überforderung infolge permanent empfundenen Zeit- und Leistungsdrucks.

Einigen Leser mag das vielleicht übertrieben vorkommen, allerdings sollte man immer bemüht sein, alle Belastungsmomente konsequent zu vermindern sowie die Motivation für den Umgang mit Ihrem Programm zielgerichtet zu erhöhen. Deshalb folgen einige grundsätzliche Betrachtungen zur Dialoggestaltung, da sie im Rahmen eines flexiblen Suchmoduls eine große Rolle spielt.

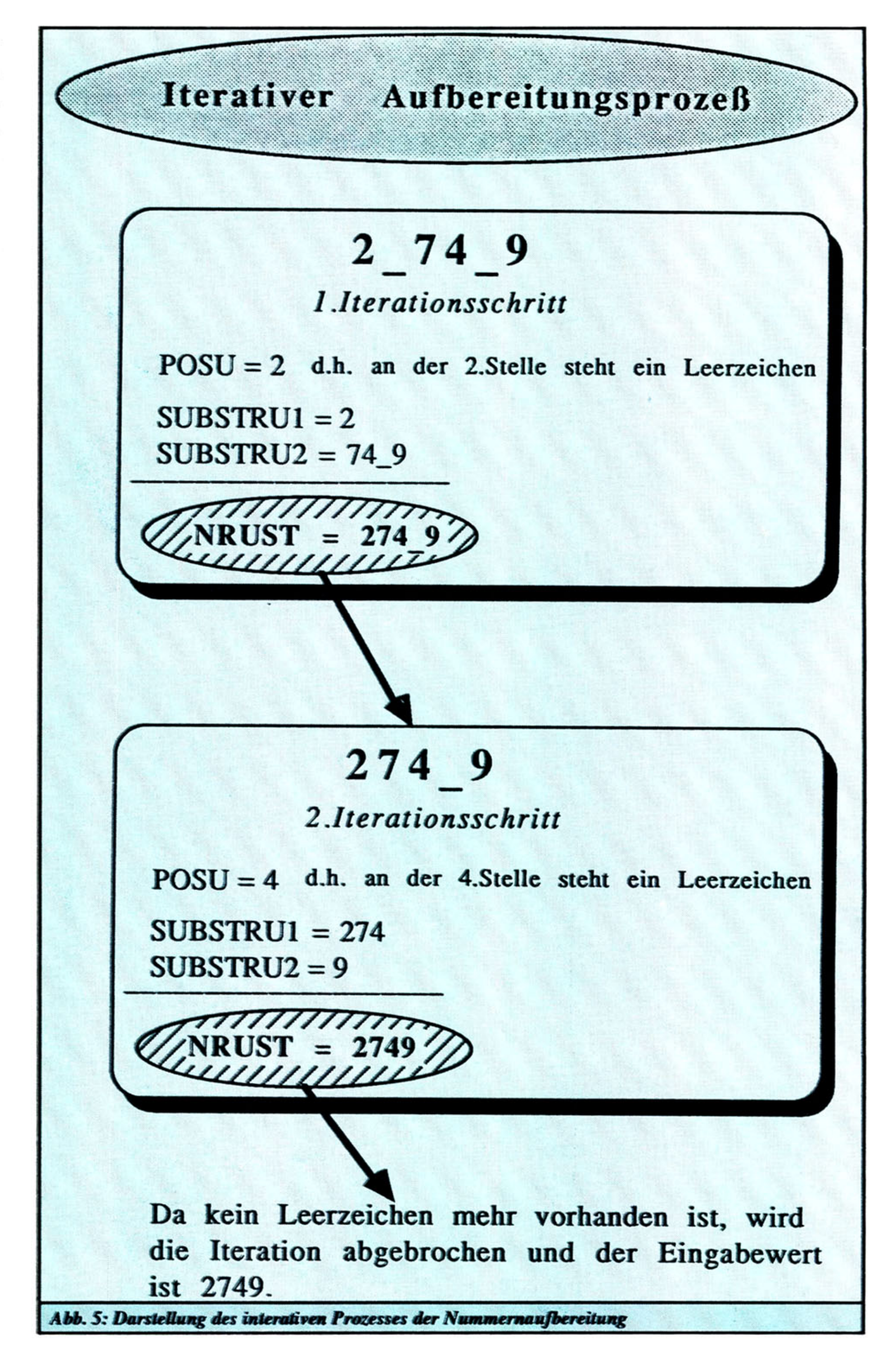


Abb. 5: Darstellung des interativen Prozesses der Nummernaufbereitung

Grundsätze der Dialoggestaltung

Eine Orientierungshilfe für Ihre Programmodulentwicklung stellt die DIN 66234, Teil 8 " Bildschirmarbeitsplätze, Grundsätze der Dialoggestaltung" dar. In dieser Norm wurden fünf Kriterien entwickelt, die eine Orientierung für Ihren Gestaltungsprozeß darstellen: - Aufgabenangemessenheit, - Selbstbeschreibungsfähigkeit, - Steuerbarkeit, - Erwartungskonformität, - Fehlerrobustheit. Diese Kriterien sind sehr allgemein gehalten und wie folgt definiert:

- Ein Dialog ist aufgabenangemessen, wenn er die Erledigung der Arbeitsaufgabe des Benutzers unterstützt, ohne ihn durch Eigenschaften des Dialogsystems unnötig zu belasten.
- Ein Dialog ist selbstbeschreibungsfähig, wenn dem Benutzer auf Verlangen Einsatzzweck sowie Leistungsumfang des Dialogsystems erläutert werden können und wenn jeder einzelne Dialogschritt unmittelbar verständlich ist oder der Benutzer auf Verlangen zum jeweiligen Dialogschritt entsprechende Erläuterungen erhalten kann.
- Ein Dialog ist steuerbar, wenn der Benutzer die Geschwindigkeit des Ablaufs sowie die Auswahl und Reihenfolge von Arbeitsmitteln oder Art und Umfang von Ein- und Ausgaben beeinflussen kann.
- Ein Dialog ist erwartungskonform, wenn er den Erwartungen der Benutzer entspricht, die sie aus Erfahrungen mit bisherigen Arbeitsabläufen bzw. aus der Benutzerschulung mitbringen oder die während der Benutzung des Dialogsystems und im Umgang mit dem Benutzerhandbuch entstehen.
- Ein Dialog ist fehlerrobust , wenn trotz erkennbar fehlerhafter Eingaben das beabsichtigte Arbeitsergebnis mit minimalem oder ohne Korrekturaufwand erreicht wird. Dazu müssen dem Benutzer die Fehler zum Zwecke der Behebung verständlich gemacht werden.

Funktionsbeschreibung des Suchmoduls anhand eines Beispiels

Nachdem Sie den Punkt 3 " Suchen von Lieferanten" angewählt haben, sehen Sie die in Abb. 2 dargestellte Bildschirmmaske. Nach den oben beschriebenen Möglichkeiten der Datensatzsuche wird eine differenzierte Auswahl der Lieferantennummmer durch die Eingabe d gewählt.

Die differentielle Eingabe ist völlig unabhängig von den Leerzeichen, die links oder rechts stehen. Die Eingabe wird immer als "d" erkannt. Die Datensätze sollen in einem von Ihnen gewählten Bereich nach den angegebenen Merkmalen durchsucht werden . Zur Hilfestellung werden dabei die letzte Lieferantennummer und die Anzahl der vorhandenen Datensätze der Datei angezeigt. Nach der Eingabe und dem Abschließen mit der Taste >Return< wird dabei die aktuelle Datensatznummer angezeigt. So weiß man zu jedem Zeitpunkt, welcher Datensatz gerade bearbeitet wird. Wenn mehrere Datensätze mit gleichen Bedingungen auftreten, werden sie sequentiell abgearbeitet. Dies ist bei größeren Datenmengen keine sehr schnelle Suche, da jedesmal auf die Datei zugegriffen werden muß. Deshalb wird bei Eingaben, die besagen, daß mehr als hundert Datensätze durchgesucht werden sollen, noch eine Sicherheitsabfrage verlangt. Dabei wird eine mittlere Zugriffszeit zugrunde gelegt und die bei der Anzahl der Datensätze geschätzte Zeit mit angegeben.

Wird in der Eingabemaske für die Lieferantennummer die Zahl 0 eingegeben, oder werden keine Angaben gemacht, wird das Programmodul beendet. Selbstverständlich erscheint vorher noch eine Sicherheitsabfrage.

In unserem Beispiel soll im Bereich von 4 bis 7 gesucht werden. Da die letzte Lieferantennummer mit der Anzahl der Datensätze übereinstimmt, wurden genau 4 Datensätze gefunden (s. Abb. 3). Nun hat man die Möglichkeit, verschiedene Optionen mittels der Funktionstasten auszuwählen.

- Es können die gefundenen Datensätze auf dem Bildschirm angezeigt werden. Dabei kann mit F01 die Adresse und mit F02 die Telefonnummer angezeigt werden.
- Es können mit F05 alle gefundenen Datensätze gedruckt werden.
- Es können mit F06 nur bestimmte Da-. tensätze gedruckt werden.
- Der Drucker kann mit F07 initialisiert werden und zuletzt ist mit der Funktionstaste F10 der Abbruch des Programmoduls möglich.

Diese einzelnen Funktionen werden mit individuellen Programmodulen erfüllt, die im nächsten Teil erklärt werden. Selbstverständlich können Sie anstatt der Telefonnummern auch nach anderen Bedingungen suchen.

PROGRAMM NAME : LSUCHE.TLK

1 Schalterstellungen / Datenbank & Datei öffnen / Bildschirm löschen

2 Ausgabe der Kopfzeile auf dem Bildschirm

lokale Variablendekl. MSG = " ", NR = "1 "

DO WHILE NR nicht 0 und MSG nicht "E" (ENDE)

lokale Variablendekl. PLAUS = "J"

DO WHILE PLAUS = "J"

3 lokale Variablendekl. VLNAME = SPACE(15), NR =SPACE(6) NRUST,NROST = "0 ", NRU,NRO = 0 MSG ="", PLAUS ="N" ,LAUF,LASTKNR, FM = 0

Datensätze zählen ⟶ DS, INDIZIEREN nach Liefernr.

LAST, Bildschirmaufbau der Maske, Einlesen der L.-NR. und des Namen des Lieferanten

SPRUNG INS UNTERMODUL LNRBER.TLK

IF NRU>LASTKNR ? JA / NEIN

JA: SPEICHERE NRU = 0,NRUST = "0 " OK = TRUE,LAUF =2, VLNAME= "", FM =1

NEIN: IF NRUST nicht null ? JA / N

JA: CASE ? 1 / 2 / OT.W.

1: SUCHE nach dem Lieferanten Namen.

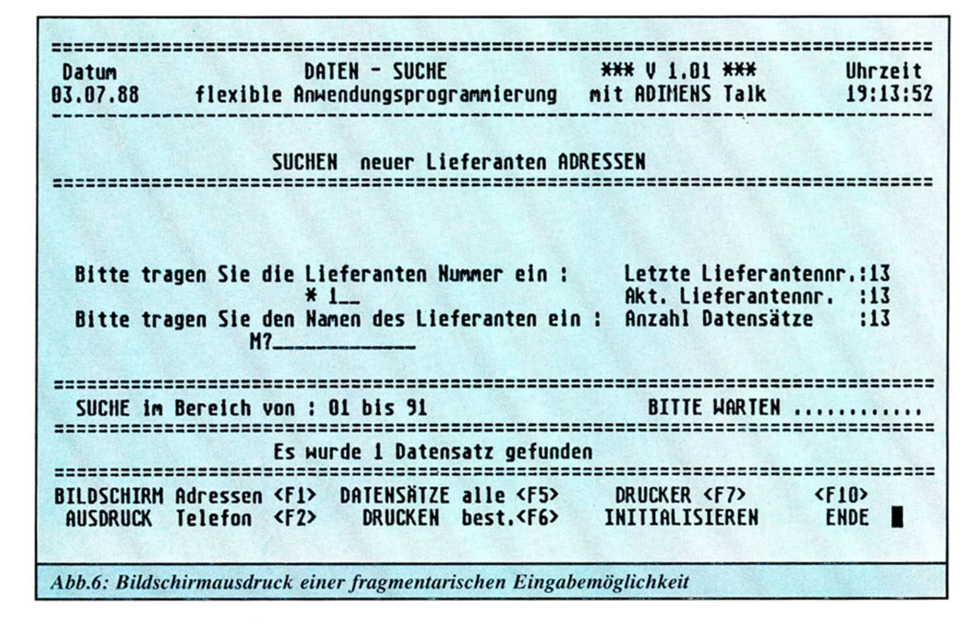
2: SUCHE nach der Lieferanten Nummer

OT.W.: Suche im Bereich von

IF alle Datensätze anzeigen lassen ? JA / N

5 SPEICHERE VLNAME = "", LAUF = "2"

Abb. 7: Struktogramm zum Modul LSUCHE.TLK - Teil 1


Datum DATEN - SUCHE *** V 1.01 *** Uhrzeit
03.07.88 flexible Anwendungsprogrammierung mit ADIMENS Talk 19:13:52

SUCHEN neuer Lieferanten ADRESSEN

Bitte tragen Sie die Lieferanten Nummer ein : Letzte Lieferantennr. :13
* 1__ Akt. Lieferantennr. :13
Bitte tragen Sie den Namen des Lieferanten ein : Anzahl Datensätze :13
M?__________

SUCHE im Bereich von : 01 bis 91 BITTE WARTEN ....

Es wurde 1 Datensatz gefunden

BILDSCHIRM Adressen <F1> DATENSÄTZE alle <F5> DRUCKER <F7> <F10>
AUSDRUCK Telefon <F2> DRUCKEN best.<F6> INITIALISIEREN ENDE

Abb.6: Bildschirmausdruck einer fragmentarischen Eingabemöglichkeit

Der Vollständigkeit halber will ich Ihnen noch zwei andere Eingabekombinationen vorstellen, bevor das Listing erklärt wird. Bei der Lieferantennummer kann beispielsweise die in Abb. 4 dargestellt Eingabe gemacht werden. Um diese Eingabe als 11 interpretieren zu können, wird ein Berechnungsmodul benötigt, das die Eingabe der Lieferantennummer aufbereitet. Ebenso wird auch bei der Eingabe eines fragmentarischen Wertes von "23 4*" das Leerzeichen zwischen der Zahl 3 und der Zahl 4 ignoriert. Im Berechnungsmodul werden die Leerzeichen entfernt, indem man das Leerzeichen lokalisiert, danach den Eingabewert in zwei Strings aufspaltet und die Leerzeichen rechts und links wegtrimmt. Danach fügt man die beiden Strings wieder zusammen. Dieser iterative Prozeß ist in Abb. 5 noch einmal dargestellt.

Hier würde der Bereich der fragmentarischen Eingabe lauten: Suche im Bereich von 2340 bis 2349. In Abb. 6. wird im Bereich von 1 bis 91 nach allen Lieferanten gesucht, die mit dem Buchstaben M beginnen. Die Suche wird automatisch beim Ende der letzten Lieferantennummer beendet.

Vorsicht ist bei der Eingabe "*" geboten. Es soll dabei im Bereich 0 bis 9 gesucht werden. Die Eingabe 0 wird aber als Bedingung für das Ende der Suche verstanden.

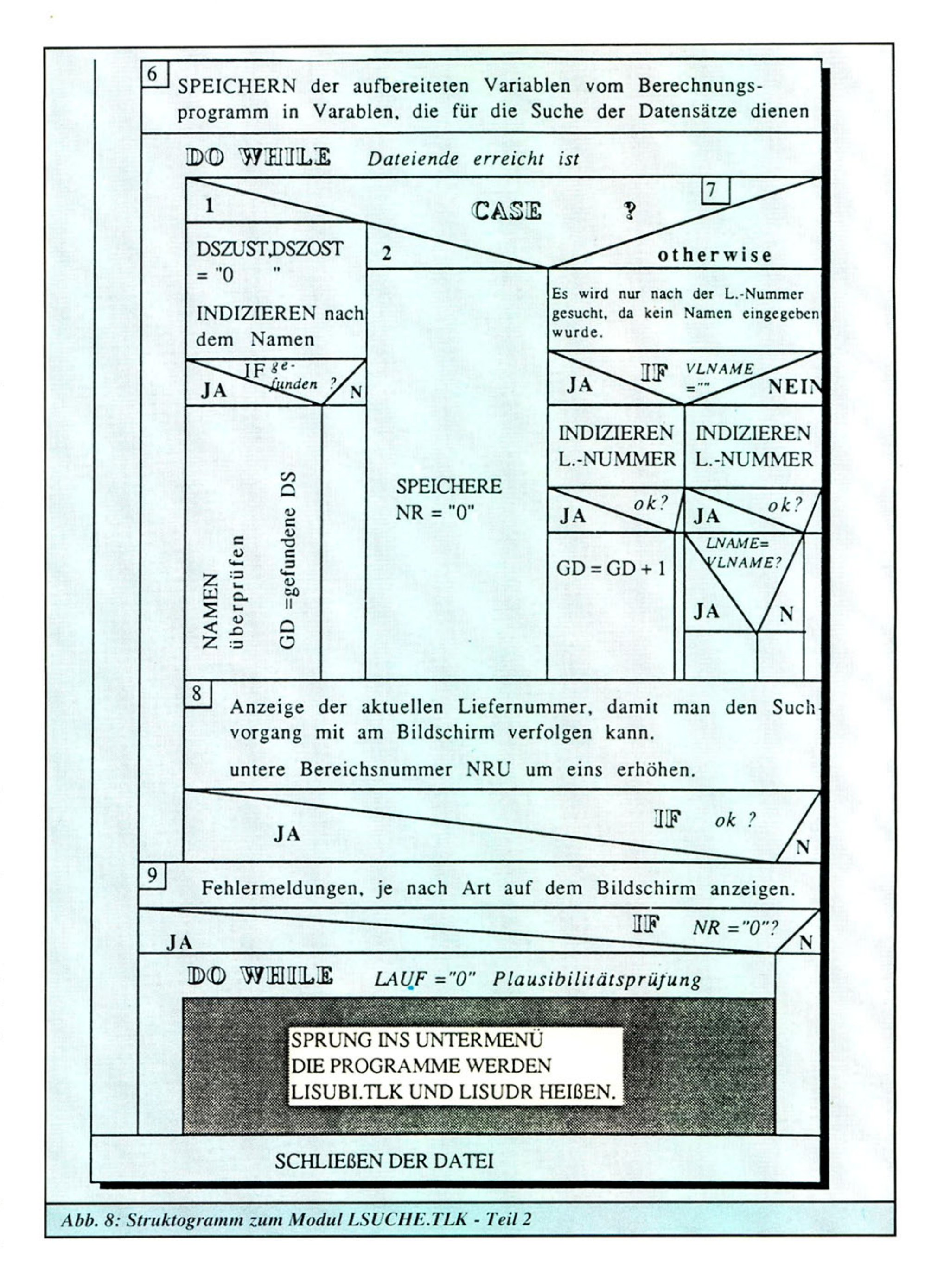


Abb. 8: Struktogramm zum Modul LSUCHE.TLK - Teil 2

Modulbeschreibung

In den Teilen 1 bis 3 wurden jedesmal sukzessive die einzelnen Kommandos zur On Line-Verarbeitung beschrieben. Da keine grundlegenden neuen Befehle hinzugekommen sind, erfolgen keine weiteren Erklärungen der ADIMENS Talk-spezifischen Befehle mehr, sondern es werden nur die Funktionen und die Anwendungen der einzelnen Moduln beschrieben. Die Details finden Sie dann in den Struktogrammen oder in den gut kommentierten Programmlistings. Über die Befehle zur reinen Datenbankmanipulation hinaus bietet ADIMENS Talk eine Reihe weiterer Befehle, die eine Programmierung auf einem hohen Sprachniveau erlauben.

Ich bitte Sie deshalb um Verständnis, daß in diesen komplexen Listings nicht alle Feinheiten ausführlich erklärt werden. Ich bin aber dennoch sehr zuversichtlich, daß Sie aus der Dokumentation einige Tips übernehmen können. Da die in der Praxis auftretenden Sonderfälle zu berücksichtigen sind, haben sich die Programme sehr rasch ausgeweitet. Die Programmroutinen zur Plausibilitätskontrolle von Dateneingaben oder zur Sicherung der Datenbestände sind deshalb genauer aus den Listings zu ermitteln.

Abb. 7 und 8 zeigen das Struktogramm des Suchmoduls, und in Abb. 9 finden Sie das Struktogramm des Berechnungsmoduls.

Zusammenfassung

Ich glaube, in diesem Teil haben Sie erkannt, wie komfortabel und elegant man spezifische Problemstellungen im Bereich des Suchens von Datensätzen lösen kann. Im nächsten Teil wird gezeigt werden, wie die Daten sortiert, aufbereitet, auf dem Bildschirm und auf dem Drucker ausgegeben werden können. Sie werden ebenfalls sehen, wie sich bei entsprechender Anpassung ihrer Variablen die Programmroutinen oder bestimmte Hilfsfunktionen sehr leicht in andere Programme integrieren lassen.

Also bis zum nächsten Mal.

Hans-Ulrich Mayer

Quellen:
DIN 66234,
Teil 8 Grundsätze der Dialoggestaltung

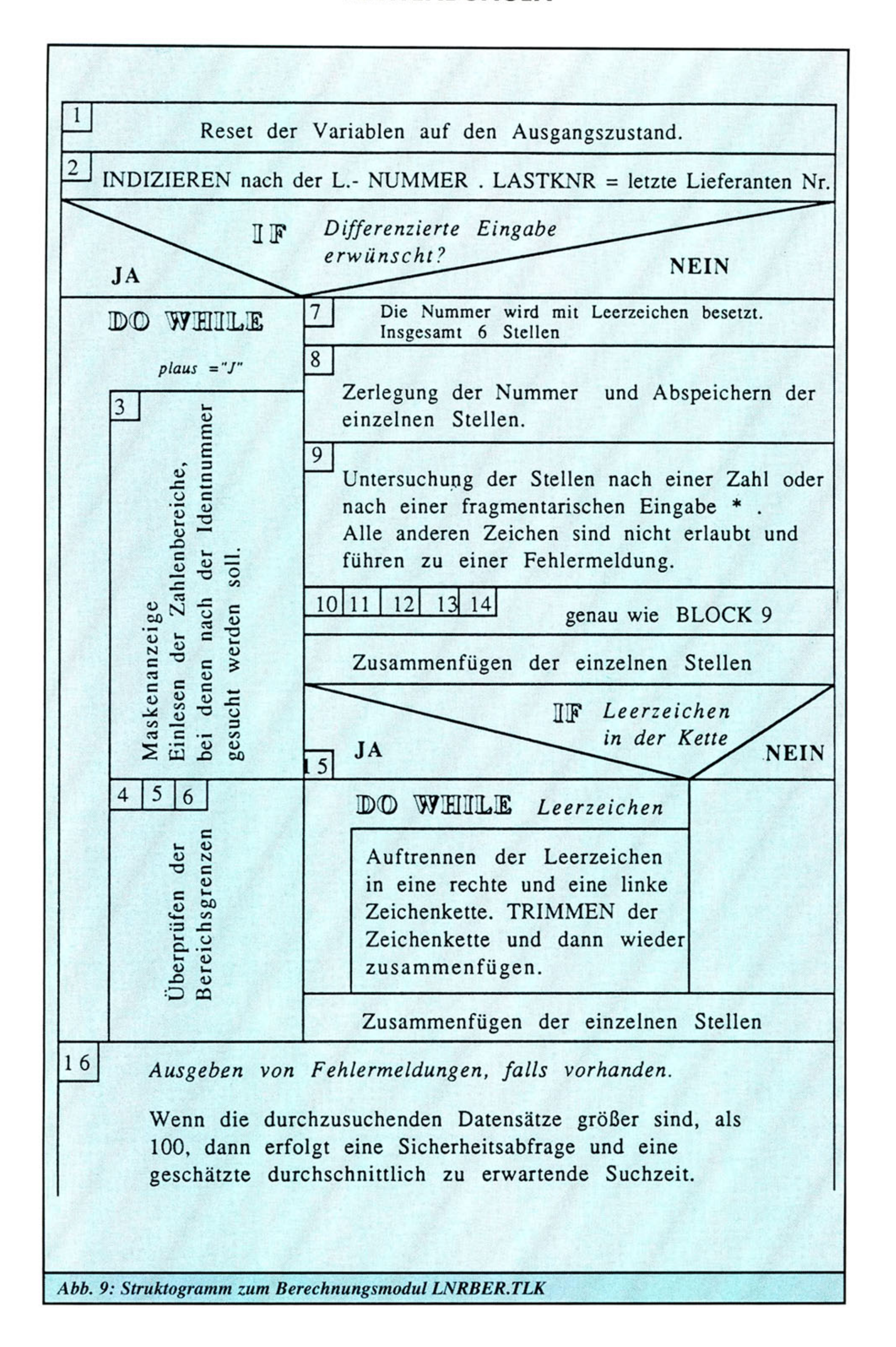


Abb. 9: Struktogramm zum Berechnungsmodul LNRBER.TLK

```
 1:  /*                  PROGRAMM LSUCHE.TLK                  */
 2:  /*********************************************************/
 3:  /*   Dies ist ein Beispiel für ein Programmodul zum     */
 4:  /*          Suchen von erfassten Datensätzen             */
 5:  /* Das Programm greift auf die Datenbank BEISPIEL zu.*/
 6:  /*         copyright by Hans-Ulrich Mayer                */
 7:  /*********************************************************/
 8:
 9:  /**** 1 ****/
10:  /* FESTLEGUNG DER SCHALTERSTELLUNGEN                     */
11:  /* Löschen des Bildschirmes                              */
12:  CLEAR
13:  /* Unterdrücken der automatischen Maskenanzeige          */
14:  SET AUTOSC OFF
15:  /* Die Sonderregelungen für die Funktion FIND sind
                                              abgeschaltet */
16:  SET DBASE OFF
17:  /* DATENBANK ÖFFNEN                                      */
18:  OPEN "BEISPIEL"
19:  /* DATEI ÖFFNEN                                          */
20:  USE ADRESSEN
21:  /**** 1 ****/
22:
23:  /*********************************************************/
24:
25:  /**** 2 ****/
26:  /* MENÜAUSGABE - Aufbau der Kopfzeile                    */
27:  CLEAR
28:  /* aktuelle Datumsanzeige und der Uhrzeit                */
29:  @ 02     ,00  SAY " Datum "
30:  @ 03     ,00  SAY DATE()
31:  @ 02     ,72  SAY "Uhrzeit"
32:  @ 03     ,72  SAY TIME()
33:  @ 01     ,00  SAY BILD1 + BILD1
34:  @ ROW()  ,TAB SAY "               DATEN - SUCHE
          *** V 1.01 *** "
35:  @ ROW()+1,TAB SAY "flexible Anwendungsprogrammierung
         mit ADIMENS Talk"
36:  @ ROW()+1,00  SAY BILD2 + BILD2
37:  @ ROW()+1,12  SAY "SUCHEN neuer Lieferanten ADRESSEN "
38:  @ ROW()+1, 00 SAY BILD1 + BILD1
39:  /**** 2 ****/
```

```
40:
41:  /****************************************************/
42:  /* Lokale Variablendeklaration                      */
43:  STORE SPACE (1) TO MSG
44:  STORE "1     " TO NR
45:  /****************************************************/
46:
47:  /****************************************************/
48:  /* Durchlaufe die Schleife 1 SOLANGE die Variablen  */
49:  /* NR<> 0 oder MSG<> "E". Bei einer Lieferanten     */
50:  /* Nummer von 0 wird das Suchen abgebrochen         */
51:  DO WHILE ( UPPER(MSG) <> "E" )  OR ( INT(NR) <> 0 )
52:  /****************************************************/
53:
54:      /**** 3 ****/
55:      STORE "J" TO PLAUS
56:
57:      /**********************************************/
58:      /* Durchlaufe die Schleife SOLANGE die Eingabe
             richtig ist.         */
59:      DO WHILE ( PLAUS = "J" )
60:      /**********************************************/
61:         /* Ab dem Mittelstück soll der Bildschirm
                                        gelöscht werden  */
62:         @ 10,00 CLEAR
63:         /* Lokale Variablendeklaration              */
64:         STORE SPACE(15)  TO VLNAME
65:         STORE "       "  TO NR
66:         STORE "0      "  TO NRUST, NROST
67:         STORE 0          TO NRU  , NRO
68:         STORE SPACE(1)   TO MSG
69:         STORE "N"        TO PLAUS
70:         STORE 0          TO LAUF
71:         STORE 0          TO LASTLNR
72:         STORE 0          TO FM
73:         /* Textaufbau für das Mittelstück */
74:         /* Zuerst soll die höchste Lieferanten Nummer
                                        ermittelt werden */
75:         INDEX LNUMMER
76:         LAST
77:         @ 11,52 SAY "Letzte Lieferantennr.:" + TRIM
                    (STR(LNUMMER,6,0) )
            /* Ermittlung der aktuellen Anzahl der Datensätze
78:                                           in der Datei */
79:         COUNT ALL TO DS
80:         @ 13,52 SAY "Anzahl Datensätze    :" + TRIM
                                            (STR(DS,6,0) )
81:         /* Textaufbau der Auswahl Maske                */
82:         @ 11,02 SAY "Bitte tragen Sie die Lieferanten
               Nummer ein : "
83:         @ 12,02 SAY "                        " GET NR
84:         @ 13,02 SAY "Bitte tragen Sie den Namen des
                                        Lieferanten ein : "
85:         @ 14,02 SAY VLNAME GET VLNAME
86:         @ 16,00 SAY BILD1  + BILD1
87:         @ 17,15 SAY "      Bedeutungen bei fragmentalem
               Suchen "
88:         @ 18,15 SAY "'*' z.Bsp 2*** Suche im Bereich von
               200 bis 299"
89:         @ 19,15 SAY "Bei der Nummer werden die
                                   Leerzeichen ignoriert. "
90:         @ 20,15 SAY "'?' z.Bsp. May? Suche Namen, die mit
               May beginnen"
91:         @ 21,00 SAY BILD1 + BILD1
92:         @ 22,00 SAY "Lieferanten Nr.= 0-> ENDE;
               Lieferanten Nr.= D->"
93:         @ 22,50 SAY "DIFFERENZIERTE BEREICHSEINGABE"
94:         /* Einlesen der Merkmalsausprägungen */
95:         READ
96:         @ 17,00 CLEAR
97:         /**** 3 ****/
99:         /********************************************/
100:        /* SPRUNG ins Untermodul LNRBER.TLP bei dem der
                                        Bereich der Nummer*/
101:        /* aufbereitet wird. Parameter Übergabe an LNRBER
                                                   : NR  */
102:        /* Parameter Übergabe zurück: NR, NRU, NRO, NRUST,
                                        NROST, LASTLNR */
103:        DO "LNRBER"
104:        /********************************************/
105:
106:        /**** 4 ****/
107:        @ 16,00 CLEAR
108:        @ 16,00 SAY BILD1 + BILD1
109:        /*Wenn die untere Bereichsnummer gröPer als die
                                     letzte Lieferanten*/
110:        /*Nummer ist, werden die Variablen für eine
                                 Fehlermeldung gesetzt.*/
111:        IF NRU > LASTLNR
112:           STORE 0     TO NRU
113:           STORE "0"   TO NRUST
114:           STORE TRUE  TO OK
115:           STORE 2     TO LAUF
116:           STORE ""    TO VLNAME
117:           STORE 1     TO FM
118:        ENDIF
119:        /* Fall, bei dem die untere Bereichsnummer nicht
                                       den Wert 0 hat.  */
120:        IF ( NOT(TRIM(NRUST) = "0" ))
121:           /* Anzeige, nach welchen Kriterien gesucht
                                                 wird. */
122:           DO CASE
123:              /* FALL 1: Suche nach dem Lieferanten
                                            Namen  ?  */
124:              CASE ( TRIM(NRUST) = "" )
125:                 IF NOT ( TRIM(VLNAME) = "" OR ASC
                                  ( TRIM(VLNAME) ) = 63 )
126:                    @ 17,02 SAY "SUCHE nach dem
                               Lieferanten Namen : "+VLNAME
127:                 ENDIF
128:              /* FALL 2: Suche nach der Lieferanten
                                               Nummer  */
129:              CASE ( INT(NRUST) = INT(NROST) )
130:                 @ 17,02 SAY "SUCHE nach der Lieferanten
                       Nr.: "+TRIM(NROST)
131:              /* ANDERENFALLS                          */
132:              OTHERWISE
133:                 STORE "SUCHE im Bereich von : " TO TEXT
134:                 @ 17,02 SAY TEXT + TRIM(NRUST) + " bis "
                       + TRIM(NROST)
135:           ENDCASE
136:        ENDIF
137:        @ 18,00 SAY BILD1 + BILD1
138:        @ 20,00 SAY BILD1 + BILD1
139:        /**** 4 ****/
140:
141:     /**********************************************/
142:     /* SchliePen der Schleife dieser Plausibilitäts
                                             kontrolle  */
143:     ENDDO
144:     /**********************************************/
145:
146:     /**** 5 ****/
147:     /* Untenstehende Kombination ist nicht sinnvoll.  */
148:     /* Wenn die Namen egal sind, so sollte man einen  */
149:     /* Suchbereich oder eine Nummer angeben. Man muP  */
150:     /* hier mit der ASC Funktion überprüfen, ob ein   */
151:     /* Fragezeichen vorliegt. Da sonst ADI Talk dafür */
152:     /*  alle Buchstaben interpretiert.                 */
153:     IF (LEN(TRIM(VLNAME))=1 AND ASC (TRIM(VLNAME))=63
                                    AND TRIM (NRUST)= "")
154:        STORE "" TO VLNAME
155:        STORE 2  TO LAUF
156:     ENDIF
157:     /**** 5 ****/
158:
159:     /**********************************************/
160:
161:     /**** 6 ****/
162:     /* Wert Übertragung der Variablen vom
                            standardisierten Untermodul  */
163:     STORE NRUST TO DSZUST
164:     STORE NROST TO DSZOST
165:     STORE NRU   TO DSZU
166:     STORE NRO   TO DSZO
167:     STORE 0     TO GD
168:     STORE FALSE TO OK
169:     STORE FALSE TO TEMP
170:     /**** 6 ****/
171:
172:     /**********************************************/
173:     /* SOLANGE SUCHEN bis das Ende der Datei erreicht*/
174:     /* ist, oder die Lauf Variable die obere Nummer  */
175:     /* des Zahlenbereiches erreicht hat.             */
176:     DO WHILE ( NOT( (NRO + 1 - NRU) = 0 OR EOF ) )
177:     /**********************************************/
178:
179:        /**** 7 ****/
180:        /* Verschiedene Fälle  der Suche nach den
                                 eingegebenen Kriterien  */
181:        DO CASE
182:           /* FALL 1: Bei Eingabe eines Leerzeichens wird
                                          die Lieferanten */
183:              /* Nummer ignoriert und es wird nach dem
```

```
                                       Namen gesucht.*/
184:      CASE (TRIM(NRUST) = "" AND TRIM (VLNAME)<> "" )
185:         @ 17, 54 SAY "BITTE WARTEN ..........."
186:         STORE "0      " TO DSZUST, DSZOST
187:         INDEX LNAME
188:         STORE FIND VLNAME TO OK
189:         IF ( OK )
190:            COUNT WHILE LNAME = VLNAME TO GD
191:         ENDIF
192:      /* FALL 2 : Bei Eingabe der Zahl 0 wird die
                                      Suche beendet. */
193:      CASE ( NRU = 0 )
194:         STORE "0" TO NR
195:      /* SONST : Überprüfen des gültigen
                                   Kundenbereiches  */
196:      OTHERWISE
197:         @ 17,54 SAY "BITTE WARTEN ..........."
198:         /* Wenn der Name nicht eingegeben wurde   */
199:         /* wird nach der Nummer des Lieferanten   */
200:         /* gesucht. Dies ist eine schnelle Suche,*/
201:         /* da nicht sequentiell nach den          */
202:         /* Datensätzen gesucht werden muP         */
203:         IF ( VLNAME = "" )
204:            INDEX LNUMMER
205:            STORE FIND INT(NRUST) TO OK
206:            /* Wenn ein Datensatz mit dieser       */
207:            /* Nummer gefunden wurde, so wird      */
208:            /* die Variable GD = Gefundene         */
209:            /* Datensätze gesetzt                  */
210:            IF (OK)
211:               STORE GD+1 TO GD
212:            ENDIF
213:         ELSE
214:            /*Sonst wird erst schnell nach der     */
215:            /*Lieferantennr. gesucht.  Wurde die   */
216:            /*Lieferanten Nummer gefunden, so      */
217:            /*wird kontrolliert, ob auch der Name */
218:            /* des Lieferanten übereinstimmt.      */
219:            INDEX LNUMMER
220:            STORE FIND INT(NRUST) TO OK
221:            IF ( OK )
222:               IF ( LNAME = VLNAME )
223:                  STORE GD+1 TO GD
224:               ELSE
225:                  STORE NOT( OK ) TO OK
226:               ENDIF
227:            ENDIF
228:         ENDIF
229:      /* AbschliePen der einzelnen Suchfälle */
230:      ENDCASE
231:      /**** 7 ****/
232:
233:      /******************************************/
234:
235:      /**** 8 ****/
236:      /* Anzeige der aktuellen Lieferanten Nummer
                                 (nur bei der Suche)*/
237:      IF NOT (TRIM (NRUST) = "" OR NR = "0" )
238:         @ 12,52 SAY "Akt. Lieferantennr.  :" + TRIM
               (NRUST)
239:      ENDIF
240:      STORE NRU + 1        TO NRU
241:      STORE STR(NRU,6,0) TO NRUST
242:      /* Merker, ob überhaupt einmal ein Datensatz
            gefunden wurde    */
243:      IF OK
244:         STORE 1 TO LAUF
245:      ENDIF
246:      /**** 8 ****/
247:
248:   /* AbschliePen der Schleife bei beendeter
                                   Datensatz Suche. */
249:   ENDDO
250:
251:    /*********************************************/
252:
253:   /**** 9 ****/
254:   /* Fehlermeldungen anzeigen mit den eingegebenen
                                            Werten  */
255:   DO CASE
256:      /* FALL 1 :                                  */
257:      CASE ( FM = 1 )
258:         BELL
259:         @ 19,00 SAY "Die Lieferanten Nr. ist
               kleiner als der zu such"
260:         @ 19,47 SAY "ende Bereich !! NEUE EINGABE "
261:         @ 22,20 SAY "BITTE eine beliebige Taste
                                      drücken !!! "
262:         WAIT
263:         STORE "0" TO NR
264:      /* FALL 2 :                                  */
265:      CASE ( (NR = "0") AND (LAUF <> 2) )
266:         @ 19,17 SAY "MODUL VERLASSEN sind Sie
                                    S I C H E R ?"
267:         @ 22,17 WAIT "      ( j / n )    "TO MSG
268:         IF ( UPPER (MSG) = "J" )
269:            STORE "E" TO MSG
270:         ENDIF
271:      /* FALL 3 :                                  */
272:      CASE ( OK OR LAUF = 1 )
273:         IF GD > 1
274:            STORE " Datensätze gefunden" TO TEXT
275:            @ 19,20 SAY "Es wurden " +
                          TRIM(STR(GD,4,0)) + TEXT
276:         ELSE
277:            @ 19,17 SAY "   Es wurde 1 Datensatz
                                          gefunden "
278:         ENDIF
279:      /* FALL 4 :                                  */
280:      CASE ( LAUF = 2 )
281:         BELL
282:         @ 19,00 SAY "Diese Eingabe ist nicht  "
283:         @ 19,43 SAY "erlaubt. Geben Sie einen "
284:         @ 22,20 SAY "DIFFERENZIERTEN Bereich ein.
               BITTE eine beliebige Taste drücken!!!"
285:         WAIT
286:         STORE "0" TO NR
287:      /* ANDERENFALLS :                            */
288:      OTHERWISE
289:         BELL
290:         @ 19,20 SAY "Es wurde kein Datensatz
               gefunden "
291:         @ 22,20 SAY "BITTE eine beliebige Taste
                                      drücken !!! "
292:         WAIT
293:         STORE "0" TO NR
294:   ENDCASE
295:   /**** 9 ****/
296:
297:   /*********************************************/
298:
299:   /**** 10 ****/
300:   IF ( NOT (NR ="0") )
301:      @ 21,00 SAY "BILDSCHIRM Adressen <F1>
            DATENSÄTZE alle <F5> "
302:      @ 22,00 SAY " AUSDRUCK  Telefon  <F2>
            DRUCKEN  best.<F6> "
303:      @ 21,50 SAY " DRUCKER <F7>       <F10>"
304:      @ 22,50 SAY "INITIALISIEREN      "
305:      STORE "0" TO LAUF
306:      /******************************************/
307:      /* SOLANGE eine falsche Eingabe gemacht wird,
                                 wird wiederholt  */
308:      DO WHILE (LAUF = "0" )
309:      /******************************************/
310:         @ 22,70 WAIT "ENDE " TO MSG
311:         DO CASE
312:            /* FALL 1 :
*/
313:            CASE ( MSG = "F10" )
314:               STORE "E"  TO MSG
315:               STORE " " TO LAUF
316:            /* FALL 2 :                            */
317:            CASE ( MSG = "F01" )
318:               DO "LEER"
319:               STORE " " TO LAUF
320:            /* FALL 3 :                            */
321:            CASE ( MSG = "F02" )
322:               DO "LEER"
323:               STORE " " TO LAUF
324:            /* FALL 4 :                            */
325:            CASE ( MSG = "F05" )
326:               DO "LEER"
327:               STORE " " TO LAUF
328:            /* FALL 5 :                            */
329:            CASE ( MSG = "F06" )
330:               DO "LEER"
331:               STORE " " TO LAUF
332:            /* ANDERENFALLS                        */
333:            OTHERWISE
334:               BELL
335:               @ 19,20 SAY "FALSCHE EINGABE !!!
                    BITTE WIEDERHOLEN"
336:               STORE "0" TO LAUF
```

```
337:            ENDCASE
338:          /* Ende der Plausibilitätsschleife          */
339:          ENDDO
340:       ENDIF
341:          /**** 10 ****/
342:    /* Schleife beenden. Modul verlassen            */
343:    ENDDO
344:    /*************************************************/
345:
346:      /* Datei wieder schliePen                                  */
347:      CLOSE
```

Listing 1: Suchmodul LSUCHE .TLK

```
  1:  /*              PROGRAMM LNRBER.T                      */
  2:  /*****************************************************/
  3:  /* Dies ist ein Programm zur Plausibilitätskontrolle */
  4:  /* der fragmentarischen Eingabe einer Nummer, welche */
  5:  /* in einem alpha-numerischen Merkmalsfeld abgelegt  */
  6:  /* wurde. Eine Zahl, bei der einer Stelle das        */
  7:  /*  zwischen 0 und 9 interpretiert.                  */
  8:  /* Zeichen * steht, wird als eine Zahl in dem Bereich*/
  9:  /* Variable NR hat 6 Stellen.                        */
 10:  /* Variablen NRU und NRO geben den Zahlenbereich an  */
 11:  /* Die einzelnen Stellen werden als .                */
 12:  /*                NRUST<n>, NROST<n> bezeichnet      */
 13:  /* Die einzelnen Stellen werden als                  */
 14:  /*                ST<n> bezeichnet, mit 0 <= n >= 9  */
 15:  /*****************************************************/
 16:
 17:  /**** 1 ****/
 18:  /* Reset der Variablen auf den Ausgangszustand       */
 19:  STORE SPACE(1) TO NRUST1, NRUST2, NRUST3, NRUST4,NRUST5
        , NRUST6
 20:  STORE SPACE(1) TO NROST1, NROST2, NROST3, NROST4,NROST
        , NROST6
 21:  STORE "0      " TO NRUST,  NROST
 22:  STORE 0          TO NRU,    NRO
 23:  /**** 2 ****/
 24:
 25:  /*****************************************************/
 26:
 27:  /**** 2 ****/
 28:  /* NR wurde in LSUCHE.TLK definiert und liegt nun hier
                                                      vor. */
 29:  /* Einlesen der einz. Stellen der eingebenen Nummer   */
 30:  /* Die Leerz.werden an den beiden Enden ignoriert     */
 31:  STORE TRIM(NR) TO NR
 32:  STORE "N" TO PLAUS
 33:  /* INDIZIEREN nach der Lieferantennr.und DS-Zeiger auf
                                     die Ende Position.    */
 34:  INDEX LNUMMER
 35:  LAST
 36:  /* Die letzte Lieferanten Nummer ist abgespeichert
                                          unter LASTLNR.   */
 37:  STORE LNUMMER TO LASTLNR
 38:  /**** 2 ****/
 39:
 40:  /*****************************************************/
 41:
 42:  /* Abfrage, ob eine differenzierte Abfrage gewünscht
                                                  wurde.   */
 43:  IF ( UPPER(NR) = "D" )
 44:     STORE "J" TO PLAUS
 45:
 46:     /*************************************************/
 47:     /* Durchlaufe die Schleife, SOLANGE PLAUS = "J"   */
 48:     DO WHILE ( PLAUS = "J" )
 49:     /*************************************************/
 50:
 51:        /**** 3 ****/
 52:        /* Löschen des Eingabe Bereiches.              */
 53:        @ 10,00 CLEAR
 54:        @ 11,02 SAY "Bitte geben Sie einen Zahlenbereich
              an. "
 55:        @ 13,52 SAY "Anzahl Datensätze:
                           " + TRIM(STR(DS,6,0) )
 56:        @ 13,02 SAY "Unterer Zahlenwert : "
                              GET NRU PICTURE "######"
 57:        @ 11,52 SAY "Letzte Lieferantennr.:
                           " + TRIM (STR(LASTLNR,6,0) )
 58:        @ 14,02 SAY "Oberer Zahlenwert :  " GET NRO
              PICTURE "######"
 59:        @ 16,00 SAY BILD1 + BILD1
 60:        /* Einlesen der beiden gewünschten Zahlenber.  */
 61:        READ
 62:        /*Überprüfung nach der Plausibiltät der Eingabe*/
 63:        /**** 3 ****/
 64:
 65:        IF ( NRU > NRO ) OR ( NRU < 1 ) OR ( NRO < 1 )
 66:
 67:           /**** 4 ****/
 68:           /* Fehlermeldung ausgeben                        */
 69:           @ 17,00 CLEAR
 70:           @ 18,17 SAY "Sie haben eine unzuläPige
                     Liefernummer eingegeben."
 71:           @ 19,17 SAY "...... bitte beliebige Taste
                     drücken !!! "
 72:           WAIT
 73:           STORE "J" TO PLAUS
 74:           STORE 0   TO NRU
 75:           STORE 0   TO NRO
 76:           /**** 4 ****/
 77:
 78:        ELSE
 79:
 80:           /**** 5 ****/
 81:           STORE "N" TO PLAUS
 82:           /**** 5 ****/
 83:
 84:        ENDIF
 85:     /* Beenden der Schleife, wenn eine zuläPige L.-Nr.
                                       eingegeben wurde    */
 86:     ENDDO
 87:     /*************************************************/
 88:
 89:     /**** 6 ****/
 90:     STORE STR( NRU,6,0 ) TO NRUST
 91:     STORE STR( NRO,6,0 ) TO NROST
 92:     STORE SPACE(6)       TO NR
 93:     /**** 6 ****/
 94:
 95:  ELSE
 96:
 97:     IF ( LEN (NR) < 6 )
 98:
 99:        /**** 7 ****/
100:        STORE NR+SPACE( 6-LEN(NR) ) TO NR
101:        /**** 7 ****/
102:
103:     ENDIF
104:
105:     /**** 8 ****/
106:     /*Die Nummer wird in ihre einzelnen Stellen zerlegt
                                     und abgespeichert */
107:     STORE SUBSTR ( NR,1,1) TO ST1
108:     STORE SUBSTR ( NR,2,1) TO ST2
109:     STORE SUBSTR ( NR,3,1) TO ST3
110:     STORE SUBSTR ( NR,4,1) TO ST4
111:     STORE SUBSTR ( NR,5,1) TO ST5
112:     STORE SUBSTR ( NR,6,1) TO ST6
113:     /*Untersuchung des Zeichens nach einer Zahl 0-9    */
114:     /*und dem Zeichen "*", welches als Fragmentzeichen*/
115:     /*dient und einem Leerzeichen. Alle anderen Zei-   */
116:     /*chen sind nicht sinnvoll und auch nicht erlaubt.*/
117:
118:     /**** 9 ****/
119:     /* ZEICHEN NR 1 */
120:     IF( (ST1="*") OR ( (47<ASC(ST1)) AND (58>ASC(ST1)) )
           OR (32=ASC(ST1)) )
121:       IF NOT (ST1 ="*")
122:          STORE ST1 TO NRUST1
123:          STORE ST1 TO NROST1
124:       ELSE
125:          STORE "0" TO NRUST1
126:          STORE "9" TO NROST1
127:       ENDIF
128:     ELSE
129:       STORE "J" TO PLAUS
130:     ENDIF
131:     /**** 9 ****/
132:
```

```
133:      /*************************************************/
134:
135:     /**** 10 ****/
136:     /* ZEICHEN NR 2 */
137:     IF( (ST2="*") OR ( (47<ASC(ST2)) AND (58>ASC(ST2)) )
           OR (32=ASC(ST2)) )
138:       IF NOT (ST2 = "*")
139:          STORE ST2 TO NRUST2
140:          STORE ST2 TO NROST2
141:       ELSE
142:          STORE "0" TO NRUST2
143:          STORE "9" TO NROST2
144:       ENDIF
145:     ELSE
146:       STORE "J" TO PLAUS
147:     ENDIF
148:     /**** 10 ****/
149:
150:      /*************************************************/
151:
152:     /**** 11 ****/
153:     /* ZEICHEN NR 3 */
154:     IF( (ST3="*") OR ((47<ASC(ST3)) AND (58>ASC(ST3)) )
           OR (32=ASC(ST3)) )
155:       IF NOT (ST3 = "*")
156:          STORE ST3 TO NRUST3
157:          STORE ST3 TO NROST3
158:       ELSE
159:          STORE "0" TO NRUST3
160:          STORE "9" TO NROST3
161:       ENDIF
162:     ELSE
163:       STORE "J" TO PLAUS
164:     ENDIF
165:     /**** 11 ****/
166:
167:      /*************************************************/
168:
169:     /**** 12 ****/
170:     /* ZEICHEN NR 4 */
171:     IF( (ST4="*") OR ((47<ASC(ST4)) AND (58>ASC(ST4)) )
           OR (32=ASC(ST4)) )
172:       IF NOT (ST4 = "*")
173:          STORE ST4 TO NRUST4
174:          STORE ST4 TO NROST4
175:       ELSE
176:          STORE "0" TO NRUST4
177:          STORE "9" TO NROST4
178:       ENDIF
179:     ELSE
180:       STORE "J" TO PLAUS
181:     ENDIF
182:     /**** 12 ****/
183:
184:      /*************************************************/
185:
186:     /**** 13 ****/
187:     /* ZEICHEN NR 5 */
188:     IF( (ST5="*") OR ((47<ASC(ST5)) AND (58>ASC(ST5)))
           OR (32=ASC(ST5)) )
189:       IF NOT (ST5 = "*")
190:          STORE ST5 TO NRUST5
191:          STORE ST5 TO NROST5
192:       ELSE
193:          STORE "0" TO NRUST5
194:          STORE "9" TO NROST5
195:       ENDIF
196:     ELSE
197:       STORE "J" TO PLAUS
198:     ENDIF
199:     /**** 13 ****/
200:
201:      /*************************************************/
202:
203:     /**** 14 ****/
204:     /* ZEICHEN NR 6 */
205:     IF( (ST6="*") OR ((47<ASC(ST6)) AND (58>ASC(ST6)) )
           OR (32=ASC(ST6)) )
206:       IF NOT (ST6 = "*")
207:          STORE ST6 TO NRUST6
208:          STORE ST6 TO NROST6
209:       ELSE
210:          STORE "0" TO NRUST6
211:          STORE "9" TO NROST6
212:       ENDIF
213:     ELSE
214:       STORE "J" TO PLAUS
215:     ENDIF
216:     /**** 14 ****/
217:
218:     /*************************************************/
219:     STORE SPACE(6) TO NR
220:     /* Die Zeichen werden zu einer Zeichenkette
                                       zusammengefügt. */
221:     IF NOT (PLAUS = "J")
222:
223:        /**** 15 ****/
224:        STORE NRUST1+NRUST2+NRUST3+NRUST4+NRUST5+NRUST6
                      TO NRUST
225:        STORE NROST1+NROST2+NROST3+NROST4+NROST5+NROST6
                      TO NROST
226:        /*Es kann nun das Problem auftreten, dass        */
227:        /*jemand eine Zahl eingibt welche innerhalbder */
228:        /*Zeichenkette durch ein Blank getrennt ist.     */
229:        /*z.Bsp. 21323. Dies soll nun als 21323 inter- */
230:        /*pretiert werden. Schleife SOLANGE bis alle     */
231:        /*Leerzeichen innerhalb ignoriert wurden         */
232:        DO WHILE ( NOT( AT(SPACE(1),NRUST)=0 OR
                   AT(SPACE(1),NROST) = 0 ) )
233:       /**********************************************/
234:       /* Trimmen aller Leerzeichen rechts und links
                                   der Zeichenkette    */
235:           STORE TRIM (NRUST) TO NRUST
236:           STORE TRIM (NROST) TO NROST
237:           /* Aufteilung in zwei Zeichenketten          */
238:           STORE AT(SPACE(1),NRUST) TO POSU
239:           STORE AT(SPACE(1),NROST) TO POSO
240:           STORE LEN(NRUST) TO LENU
241:           STORE LEN(NROST) TO LENO
242:           STORE SUBSTR(NRUST,POSU+1,LENU-POSU)
                                            TO SUBSTRU1
243:           STORE SUBSTR(NROST,POSO+1,LENO-POSO)
                                            TO SUBSTRO1
244:           STORE SUBSTR(NRUST,1,POSU-1) TO SUBSTRU2
245:           STORE SUBSTR(NROST,1,POSO-1) TO SUBSTRO2
246:           STORE SUBSTRU2 + SUBSTRU1 TO NRUST
247:           STORE SUBSTRO2 + SUBSTRO1 TO NROST
248:        ENDDO
249:        /* Ende der Schleife der Leerstellen
                                        Substitution*/
250:        /*********************************************/
251:
252:        /* Bildschirmausgabe                            */
253:        STORE INT(NRUST) TO NRU
254:        STORE INT(NROST) TO NRO
255:        /***** 15 ****/
256:
257:     ENDIF
258:  ENDIF
259:
260:  /**** 16 ****/
261:  /* Es ist nur sinnvoll bei kleineren L.-Nr. als die
                                Letzte Nr. zu suchen */
262:  IF LASTLNR < NRO
263:     STORE LASTLNR TO NRO
264:  ENDIF
265:  /* Gültige Bereichsbeschränkung                      */
266:  IF ( PLAUS = "J" )
267:     BELL
268:     @ 18,17 SAY "Sie haben eine ungültige Eingabe
                                        gemacht !!!   "
269:     @ 19,17 SAY "......Bitte eine bel. Taste drücken   "
270:     WAIT
271:  ENDIF
272:  IF ( NRO - NRU ) > 100
273:     BELL
274:     @ 17,17 SAY "Ihr gewählter Bereich ist gröPer als
100. "
275:     @ 19,17 SAY "Anzahl durchsuchender Datensätze = "+
                                  TRIM(STR(NRO-NRU,6,0))
276:     /* Berechnung der erforderlichen Zeit für die
                                        Datensuche. */
277:     /* DSUZEIT ist eine Dezimalzahl -> Typ REAL          */
278:     STORE (NRO-NRU) * 0.2/60 TO DSUZEIT
279:     @ 20,17 SAY "durchschnittliche Suchdauer beträgt
                                               ca.: "
280:     IF ( DSUZEIT < 1 )
281:        /* Anzeige in Sekunden                         */
282:        @ 20,59 SAY TRIM(STR(INT((NRO-NRU)*0.2),2,0)) +
                                          "SEKUNDEN"
283:     ELSE
284:        @ 20,59 SAY TRIM(STR(INT(DSUZEIT),2,0)) + " MIN "
285:        @ 20,66 SAY STR(INT(0.2*((NRO-NRU)-INT(DSUZEIT) *
                                     300)),2,0) + " SEC "
```

```
286:        ENDIF
287:        @ 21,17 SAY  "Dies nimmt  V I E L  Z E I T  in
                                              Anspruch !!! "
288:        @ 22,17 SAY  "Sind Sie sich  S I C H E R   ???   "
289:        @ 22,59 WAIT "       (J / N)  " TO PLAUS
290:        IF (UPPER (PLAUS) = "J" )
291:           STORE "N" TO PLAUS
292:        ELSE
293:              STORE "J" TO PLAUS
294:           ENDIF
295:     ENDIF
296:     /**** 16 ****/
297:     /*****************************************************/
```

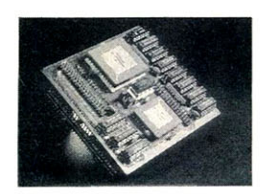
Listing 2: Berechnungsmodul LNRBER. TLK

ENDE

MARFLOW
modular systems GmbH

Tunen Sie Ihren 68000-Rechner

PAK-68



PAK-68 ist eine Prozessor-Austausch-Karte mit der 32-Bit-CPU **68020** und der FPU **68881.** PAK-68 wird in 68000-/68010-Systemen (wie Atari ST, Commodore AMIGA, mc-68000, VME-Board usw.) statt der 16-Bit-CPU eingesetzt. Ein Performance-Gewinn bis weit über 100 %* ist die Folge! Mit weiterer Leistungssteigerung durch Einsatz des Arithmetik-Coprozessors 68881. PAK-68 ist 10×10 cm „klein" und ab **DM 798,–** erhältlich.
* (je nach Software)

MARFLOW modular systems GmbH
Vahrenwalder Straße 7, 3000 Hannover 1
Telefon: (0511) 3563280
Telex: 923798 tch d, Telefax: 3563100





- Alle Programme werden auf erstklassigem Diskettenmaterial geliefert!
- PD-Software.

Jede Einzeldiskette (SS oder DS formatiert)
nur: DM 5,99,–
Preis je 10er Block DS formatiert; 5 Disketten
nur: DM 29,99,–
Preis je 10er Block SS formatiert; 10 Disketten
nur: DM 49,99,–
Senden Sie Ihre Bestellung bitte an:

Active Software
Ostpreußenstr. 4
8400 Regensburg

[ ] Nachnahme: (Nur Inland zuzügl. DM 5,- Nachnahmegebühr).
[ ] Scheck über DM .... liegt bei.
- Alle Preise inklusive Porto und Verpackung.

eine Nummer schlauer
Foxware — **CW-CHART** Börsensoftware

AKTIEN-CHARTANALYSE

ATARI Komplettsoftware für Börsianer
CW-CHART V 6.0 ★ CW-DEPOT ★ CW-BTX ★ DEMO

★ Autom. Aktienkursübernahme von BTX
★ mtl. BTX-Abrufkosten ca. DM 5,–
★ täglich akt. (K)auf- u. (V)erkaufsignale
★ Point & Figure Chart
★ Linechart, Balkenchart, Ralativchart, Vergleichchart
★ Trendindikator
★ Overbought / Oversold
★ Momentum
★ Advance / Decline Chart
★ Optionsscheinanalyse
★ TBI – RSI Chart
★ Variabler Datumsrahmen
★ flexible GD-Linien
★ unbegrenzte Aktienzahl
★ kostenloser Urlaubsservice
★ Programm incl. akt. Kursordner (ca. 300 Aktien)

Demo, Preis- u. Leistungsverzeichnis DM 10,–

Ein starkes **ANLAGEINSTRUMENT**
– ein **MUSS** für **BÖRSIANER** –

FOXWARE BUCHSTEINWEG 1 8172 LENGGRIES TEL/BTX 08042/2175

ATARI ST im *Profieinsatz*
IAS.
Im Angebot: Nur beste Hardware + Spitzenprogramme **InfoBlatt** anfordern, bitte Bereich angeben

EDV-Beratung und Einrichtung
Hard- und Software
wissenschaftliche Textverarbeitung
Video-Programmschulung

10 Markendisketten 3M double 135 TPI	21,--		
Signum!Zwei	428,--	Stad	168,--
Daily Mail	172,--	Tempus 2.0	102,--
Kaiser	119,--	Flexdisk	65,--

At-Tastatur, Fest- u. Wechselplatten
Drucker, Btx-Anschluß, weitere Programme
Büroarbeitsplatz/Schulung:ca. 10.000DM
Hotline und Service: di 18:00 - 19:00

Intelligent Applications & Services
An der Maar 5 4053 Jüchen 6
☎ **0 21 81 - 49 83 37**

ATARI ST SOFTWARE

AS-HAUSHALT (Haushaltsbuchführung) **DM 99.-**
Buchführung für alle Privathaushalte. Buchen über Konto/Gegenkonto. Suchen und Löschen von Buchungen. Auto-Save. Monats- und Jahresabschluß. Druckeranpassung. Journalausdruck. Balkengrafik. Deutsches Handbuch.

ST-FIRMENBUCH (Buchführung) **DM 139.-**
Doppelte Buchführung für kleinere Gewerbetreibende. Automatisches Mitführen von MwSt-Konten. 5 versch. MwSt-Sätze frei einstellbar. Universelle Druckeranpassung. Abschlußzeitraum Monat/Quartal/Jahr. Umsatzsteuervoranmeldung. Journalausdruck. Ausführliches deutsches Handbuch.

ST–ÜBERWEISUNGSDRUCK **DM 45.-**
Bedruckt alle Arten von Überweisungsträgern, Schecks, Zahlkarten usw. Universelle Druckeranpassung, eigene Formulare können mit einem Texteditor selbst erstellt werden. Automatische Umwandlung des Betrages in Zahlwort. Eingaben sind abspeicherbar.

DATENBANKANWENDUNGEN **DM 69.-**
6 Anwendungen für Adimens ST, z.B. Termin-Adressenverwaltung, Lager.

ST-SCHREIBMASCHINE **DM 59.-**
Mit diesem Programm arbeitet Ihr ST+Matrixdrucker wie eine Schreibmaschine. Sehr gut geeignet zum Ausfüllen von amtlichen Formularen. Ausdruck entweder direkt oder über editierbares Display. Mit dt. Handbuch.

ST-ETIKETT **DM 59.-**
Bedruckt alle Arten von Etiketten. Jede Art von Etiketten kann separat mit einem Texteditor definiert werden. Mit Seriennummerngenerierung.

ST-AUFTRAG (Fakturierung) **DM 398.-**
Eine komfortable und einfach zu handhabende Fakturierung unter GEM. Ausgabe von Listen, Serienbriefen, Etiketten sowie Fakturformularen mit 1st Word Plus Schriftattributen. Alle Formulare sind selbst anpaßbar. Demoversion mit Handbuch DM 40.- (wird angerechnet).

SOFORT KOSTENLOSE INFO 4/88 ANFORDERN

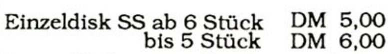
AS DATENTECHNIK
AS-DATENTECHNIK • Mainzer Str. 69
6096 Raunheim • Tel. 06142/22677

HSS
Hardware-Software Systemlösungen
Harmel-Scollar & Schmithals GbR
Darmstädter Str. 20 · 5000 Köln 1 · ☎ 0221/316207

Cash Soft	**T.I.M. (Buchführung)**	**259,–**
	Cashflow (Kassenbuch)	**259,–**
Prodata	fibuMAN e	389,–
Atari	Calamus	378,–
	1.st Word Plus & 1.st Mail	188,–
	Adimens ST/Talk	je 188,–
	SH 205	1049,–
Vortex	HDPLUS 20, 20 MB	1149,–
	HDPLUS 30, 30 MB	1298,–
	HDPLUS 60, 60 MB	2098,–
Prospero	Pro Fortran 77	358,–
	Pro Pascal	248,–
GFA	GFA Basic 3.0	178,–
	GFA Draft Plus 3.0	298,–
CCD	Tempus 2.0	109,–
Application	Stad	159,–
	Signum II	369,–
	Daily Mail	159,–
	Megamax Modula 2/Lacer C	je 349,–

Weitere Hard- und Software auf Anfrage.

Versandbedingungen: Preise inkl. Versandkosten (Scheck), bei Nachnahme zuzügl. DM 5,-, alle Preise für Atari ST-Version.



Einzeldisk SS ab 6 Stück	DM 5,00
bis 5 Stück	DM 6,00
Doppeldisk z.B. 21/22 (ungerade beginnend und aufeinanderfolgend) ab 6 Stück	DM 6,00
bis 5 Stück	DM 7,00
Pakete 1-10, 11-20, 21-30, u.s.w. auf 5 Disketten DS	DM 25,00

Preise incl. Disk zzgl. Porto u. Verp. DM 5,00, NN plus DM 1,70, besser ist ein V.-Scheck (Ausland nur V.-Scheck)

iks
Schönblickstraße 7
7516 Karlsbad 4
☎ **ab 18 Uhr 07202/6793**

LOOK

LASERAUSDRUCK für ALLE !
Atari ST Dokumente ausdrucken auf dem Laserdrucker

Texte, Grafiken, Schriftsätze usw.
Ausdruck von *OFFSET* Vorlagen
Diskette einsenden genügt!

Kostenlose Information und aktuelle Preisliste anfordern (...ab -,31 DM pro Seite)

....äußerst günstige Preise für Atari und Vortex Produkte....
...Flachbettscanner, Festplatten und Schneider Computer.

COMPUDRUCK
computerunterstützte Erstellung von hochwertigen Druckerzeugnissen

Klaus Plüher Karl-Friedrich-Friesenstr. 26 4690 Herne 1
Tel.Nr.: 02323/46362 Btx: 0232346329 0001

SCHNELLE 3D AUF DEM ST

Flächendeckende Objektgrafik in GFA-BASIC

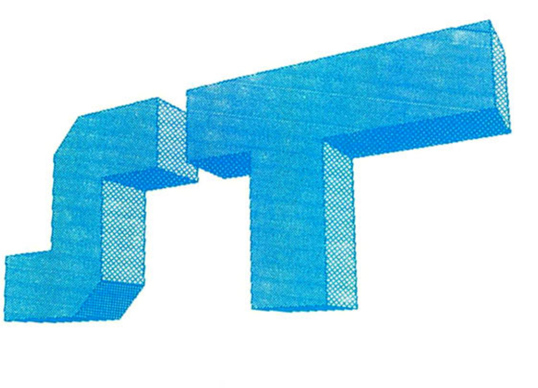
Teil I

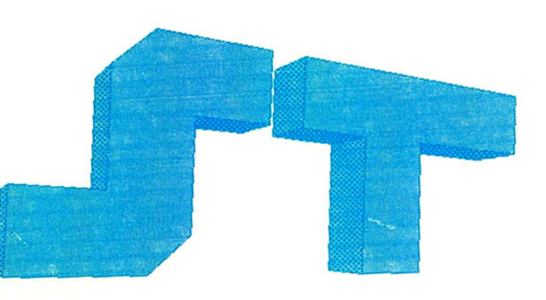
Dreidimensionale Grafiken gehören immer noch zu den beliebtesten Demoprogrammen, und gerade dieser Show-Effekt verleitet viele Computerbegeisterte zu dem Wunsch, sich doch selber einmal in der Programmierung solcher Grafiken zu versuchen. Nach flüchtigem Studium der dickleibigen Fachliteratur schüttelt manch einer verwirrt den Kopf. So schwer soll das sein? Ist das ohne ein Studium der höheren Mathematik überhaupt zu bewältigen? Nun, diese Artikelserie soll eine allgemeinverständliche Einführung in die Geheimnisse der 3D darstellen und zeigt ein Programm in GFA-BASIC, das sich gut zur Darstellung bewegter, flächendeckender 3D-Grafik eignet.

Hinter jeder Weisheit steckt ein kleiner Schaltkreis (Stefan Krawczyk).

Dreidimensionale Gegenstände können auf viele verschiedene Wege beschrieben und dargestellt werden. Begriffe wie Draht-, Flächen- und Volumenmodell geistern durch Fachliteratur und Zeitschriften. Auch ist die 3D ein dermaßen weites Feld, daß man sich von Anfang an auf ein Gebiet davon beschränken muß. Im folgenden werden wir uns nur mit Objekten auseinandersetzen, die durch feste Punkte in einem dreidimensionalen Raum beschrieben werden, also nur aus ebenen Flächen bestehen. Altbekanntes Beispiel: Der Würfel. Objekte mit gekrümmten Oberflächen (Kugel, Kegel etc.) fallen also schon aus der Definition heraus und können bestenfalls durch Annäherungen dargestellt werden (z.B. Längen- und Breitengrade eines Globus').

Auch das allseits umjubelte Ray-Tracing muß leider der Geschwindigkeit geopfert werden. Besonders die Spezialeffekte dieser Grafikspielart (Spiegelung, Transparenz, etc.) fordern viel CPU-Zeit und werden bestimmt noch viele Artikel in den Fachzeitschriften benötigen. Gewisse Ähnlichkeiten lassen sich aber nicht leugnen, die räumliche 3D-Grafik war ja der Ausgangspunkt für die Prinzipien des Ray-Tracing.

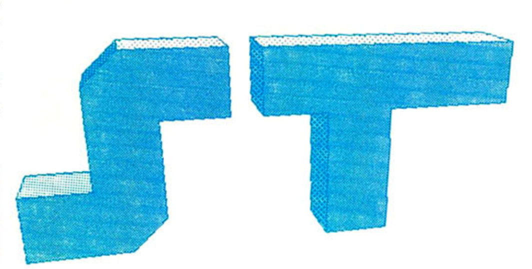




Darstellung

Die primitivste Darstellung eines Würfels anhand seiner acht Eckpunkte heißt folgerichtig Punktdarstellung und kann, wie jedem einsichtig sein müßte, mit der Variablen *Punktanzahl* und dem Feld *XYZ(Punktanzahl,3)* dem Computer übergeben werden. Die Darstellung auf dem Monitor erfolgt wie bei dem Koordinatensystem der Schulmathematik, die dritte Achse Z zeigt in die Tiefe, die Berechnung erfolgt also folgendermaßen:

```
X2D =  X3D  +  Z3D  /  XVERZERRUNG
Y2D =  Y3D  +  Z3D  /  YVERZERRUNG  (1)
```



X3D, Y3D, Z3D sind die Koordinaten eines Punkts, die X- und Y- Verzerrung gibt den Verlauf der Z-Achse an; beim ST ist ein Wert von ca. 10 empfehlenswert (Abb. 1).

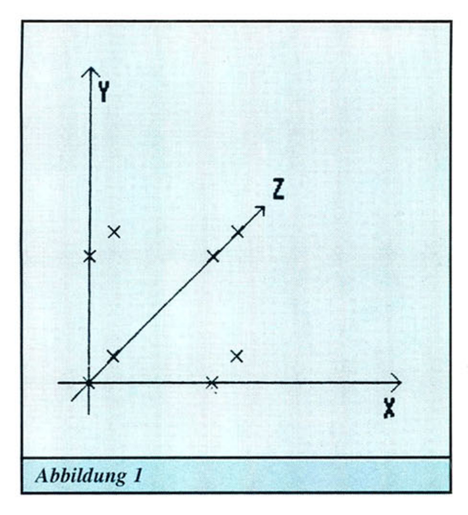
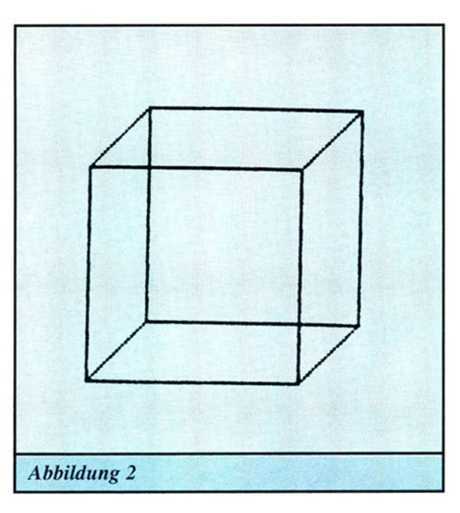


Abbildung 1

Aus dem Punktmodell folgt die schon etwas komplexere Variante, das sogenannte Drahtmodell. Hierbei werden die Linien des Objekts in einem weiteren Feld zusammengefaßt, das auf die Punkte von dem ersten Feld verweist. Diese Darstellung erzeugt den Effekt, daß die Begrenzungslinien eines Objekts wie aus Draht erscheinen, es sind also immer alle Linien sichtbar (Abb. 2).


Abbildung 2

Bei der Darstellung eines Drahtmodells nach den oben angegebenen Formeln fällt auf, daß der Würfel nicht sehr naturgetreu wirkt. Die räumliche Tiefe, die wir mit unseren zwei Augen erfassen, fehlt. Um diese Sichtweise auf einem zweidimensionalen Bildschirm nachzubilden, müssen entferntere Linien verkürzt werden, die Punkte müssen also mit zunehmendem Z näher an den Bildschirmmittelpunkt herangerückt werden. Dieses Verfahren, das unter den Begriffen perspektivische Verkürzung oder auch Zentralprojektion firmiert, ersetzt Formel (1) durch folgende Rechnung:

```
X2D = X3D * XVERZERRUNG / Z3D
Y2D = Y3D * YVERZERRUNG / Z3D (2)
```

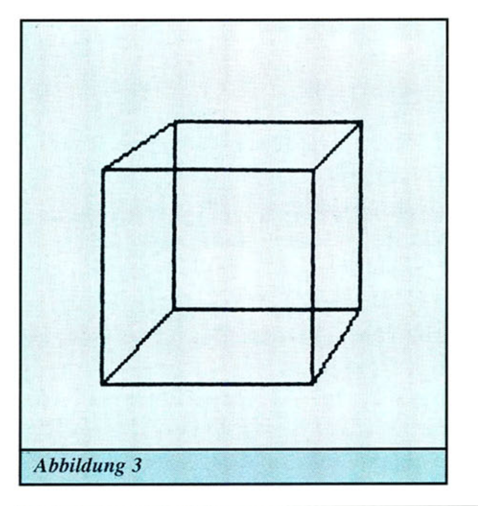
Für die Verzerrung ist jetzt ein Wert um 500 empfehlenswert. Den Unterschied können Sie an den Abbildungen 2 und 3 deutlich erkennen.

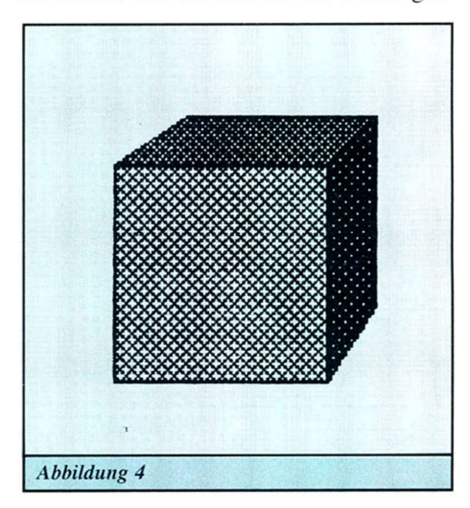
Als nächste Verfeinerung wird zumeist das Drahtmodell mit Hidden-Line-Algorithmus benutzt. Damit werden nur die Linien dargestellt, die sichtbar sind, wenn das Objekt aus einer undurchsichtigen Masse besteht. Der Hidden-Line-Algorithmus beschäftigt sich damit, herauszufinden, ob eine Linie dargestellt werden muß oder nicht. Da dies einiges an Rechen- beziehungweise Programmieraufwand benötigt und von unserem Programm eh erledigt wird verzichten wir auf die ausladende Beschreibung und leiten über zu der Variante, die das Programm benutzt, dem Flächenmodell.

Dieses Modell hat den Vorteil, daß die Objekte einen Inhalt zu haben scheinen. Es sind nur die tatsächlich sichtbaren Flächen und Begrenzungslinien zu sehen. Die Flächen erhalten dabei charakteristische Farben oder Muster, die der Wiedererkennung und dem geometrischen Vorstellungsvermögen weitaus näher kommen als das Drahtmodell (Abb. 4).

Modell

Beim Flächenmodell wird neben dem Feld der Punkte ein weiteres Feld benötigt, in dem zu jeder Fläche die Eckpunkte sowie Angaben über Farbe und Muster gespeichert werden. Aus programmiertaktischen Gründen sind jetzt einige Einschränkung und Anmerkungen nötig. Dem Programm muß ja nicht nur erklärt werden, aus welchen Punkten sich eine Fläche zusammensetzt, sondern auch, auf welcher Seite sich der Inhalt des darzustellenden Objekts befindet. Unser Gehirn stellt solche Sachverhalte so automatisch klar, daß - wie so oft - erst der Computer uns mit solchen Notwendigkeiten konfrontiert. Nun, das Problem läßt sich am leichtesten dadurch lösen, daß eine Drehrichtung der Punkteingabe festgelegt wird. Die Flächen müssen mit der sogenannten Korkenzieherregel (rechtsherum geht er in die Flasche!) definiert werden. Man muß sich immer vorstellen, vor den Punkten des Objekts zu stehen und die Fläche im Uhrzeigersinn festzulegen. Diese Vorgehensweise hat den entscheidenden Vorteil, daß sich der Programmieraufwand, festzustellen, ob eine Fläche sichtbar ist oder nicht, drastisch vereinfacht.


Abbildung 3


Abbildung 4

Hierzu wird das sogenannte Kreuzprodukt ermittelt, ein Vektor, der immer senkrecht zu der Ebene steht, aus der er berechnet wurde. Zur Berechnung genügen drei Punkte, da diese eine Ebene ja schon allgemein beschreiben. Wie in Abb. 5 gut zu sehen ist zeigen diese Vektoren wie Pfeile von den Flächen in das Objekt hinein. Und eben hier läßt sich leicht entscheiden, ob eine Fläche sichtbar ist oder nicht. Solange der Vektorpfeil von dem Betrachter wegzeigt, ist die Fläche sichtbar, falls der Pfeil ihm entgegenkommt, unsichtbar. Da die Tiefe ja durch die Z-Achse bestimmt wird, genügt es, den Z-Wert des Vektors zu ermitteln, und falls dieser positiv ist, zeigt er nach vorne, und die Fläche ist unsichtbar. Aufmerk-

```
XKREUZ = (YP3 - YP2) * (ZP2 - ZP1) - (YP2 - YP1) * (ZP3 - ZP2)
YKREUZ = (XP3 - XP2) * (ZP2 - ZP1) - (XP2 - XP1) * (ZP3 - ZP2)
ZKREUZ = (XP3 - XP2) * (YP2 - YP1) - (XP2 - XP1) * (YP3 - YP2) (3)
```

```
XTURN = X * COS(Winkel) + Y * SIN(Winkel)
YTURN = Y * COS(Winkel) - X * SIN(Winkel)                    (4)
```

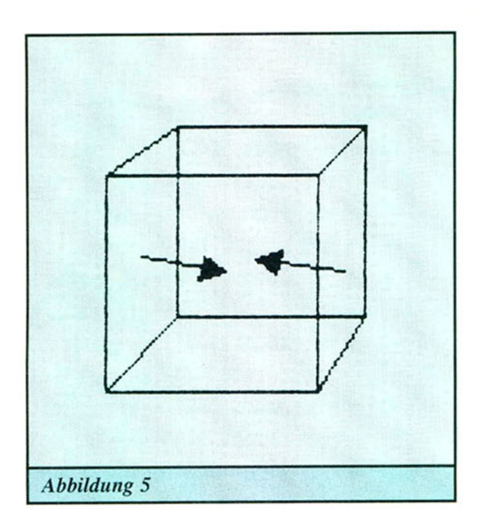


Abbildung 5

same Leser werden jetzt einwenden, daß das Kreuzprodukt ja wahlweise in die eine oder andere Richtung der Ebene zeigen kann, es steht ja in jedem Falle senkrecht zur Ebene; aber hierfür wurden die Punkte nach der Korkenzieherregel geordnet. In der richtigen Reihenfolge zeigt der Vektor immer in das Objekt hinein, also in gleicher Richtung wie der Korkenzieher. Fehlt nur noch die Formel (siehe Formel (3)).

Benötigt wird lediglich die Berechnung von ZKREUZ, die Formeln für X und Y stehen nur der Vollständigkeit halber hier. Nun soll sich unser Objekt ja nicht nur auf Geraden über den Bildschirm hin und her bewegen können. Eine Möglichkeit, das Objekt nach Belieben zu kippen und zu drehen, verhilft ihm zu größerer Anschaulichkeit. Dies ist am einfachsten durch eine Drehung an den Koordinatenachsen zu erreichen, indem die einfache Formel zum Drehen von zweidimensionalen Punkten (siehe unter Formel (4)), in das Dreidimensionale übertragen wird. Hier gibt es dann drei mögliche Drehungswinkel, die Neigung, die Richtung und die Kippung (NRK). Abbildung 6 zeigt den Sachverhalt. Durch dreimaliges Einsetzen der Formel (4) entsteht ein Formelmonster, das durch Auflösen zu den Eingangskomponenten (X,Y,Z) vereinfacht werden kann.

Zudem empfiehlt es sich, die trigonometrischen Funktionen nur einmal durchzuführen.

Also:

Diese Formeln, die leicht abgewandelt als Transformationsmatrizen bezeichnet werden, drehen ein Objekt dreidimensional um drei Winkel. Es muß aber beachtet werden, daß alle drei Winkel sich gegenseitig beeinflussen (nach einer Kippung von 90° erscheinen dem Betrachter Neigung und Richtung wie vertauscht), und daß die Drehung sich immer um den Nullpunkt vollzieht. Dieses läßt sich aber leicht ändern, indem man alle Eingangswerte (X,Y,Z) von dem gewünschten Drehungspunkt subtrahiert.

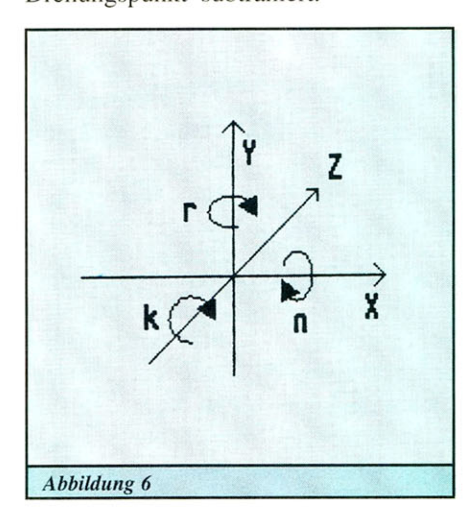


Abbildung 6

Unsere Formelsammlung ist nun schon recht komplex, und bevor wir uns mit dem letzten Problem auseinandersetzen, klären wir die Reihenfolge der Formeln. Am Anfang war das Objekt, das durch seine Koordinaten festgelegt ist. Danach kam die Drehung, also werden alle Punkte des Objekts von dem Drehungspunkt abgezogen und dann durch die Transformationsmatrix gehetzt (Formeln 5 und 6). Jetzt kann jeder Punkt einem globalen Bezugspunkt zugeordnet, sprich hinzuaddiert werden. Dies ist sehr hilfreich, wenn man die Objektposition verändern will, ohne gleich alle Koordinaten des Objekts abzuwandeln. Auch kann damit dasselbe Objekt gleichzeitig an mehreren Positionen projiziert werden. So, jetzt jagt man die Punktdaten durch die perspektivische Verkürzung (Formel 2), und erst danach kann mittels des Kreuzprodukts entschieden werden, ob die Fläche sichtbar ist oder nicht. Hierzu benutzt man die oben beschriebene Formel 3 und füttert sie mit den Koordinaten der ersten drei Punkte.

Jetzt fehlt nur noch die Darstellung, oder..?

Teufel auch ...

Tja, oder. Jetzt schlägt der Teufel im Detail zu. Bei vielen Objekten werden ja Flächen durch davorliegende verdeckt. Ein offenes Regal beispielsweise (Abb. 7).

Hier werden Flächenstücke durch andere verdeckt, und unser Projekt ist nicht im mindesten darauf eingerichtet. Dieses Problem ist auch bei vielen CAD-Programmen bekannt, und leider gibt es bis heute noch keine allgemeingültige Lösung (zumindest keine, die sich mit Geschwindigkeit kombinieren läßt).

Aufgeben, verzweifeln?

Nun, auch mit Teillösungen läßt sich viel Abhilfe schaffen. So kann mit der Sortierung der Flächen vor der Ausgabe einiges erreicht werden. Es werden die Mittelpunkte aller Flächen ermittelt und nach der Tiefe (Z-Wert) sortiert. Die entfernteste Fläche wird zuerst dargestellt, dann die nächste usw. Durch das Überzeichnen werden die Teilflächen hergestellt, und in

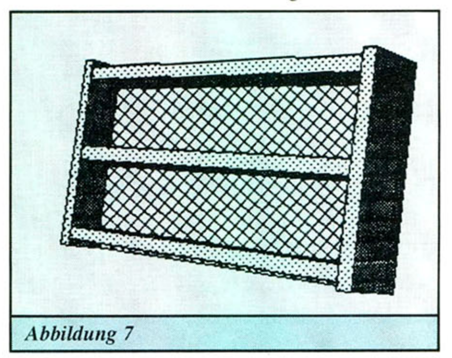
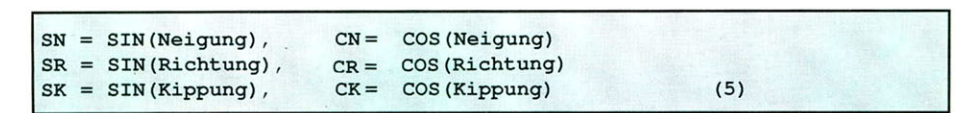


Abbildung 7

den meisten Fällen stimmt die Struktur. Das Problem tritt jetzt nur noch bei extrem schmalen und langen Flächen auf, wo der Mittelpunkt wenig darüber aussagt, welche Flächen wo verdeckt werden. Diese Fälle lassen sich vermeiden bzw. durch Aufteilen einer solchen Fläche in mehrere kleine ganz ausschalten.

Die Berechnung ist denkbar einfach; es werden alle Z-Werte der umgerechneten Punkte zusammenaddiert und durch die Anzahl der Punkte geteilt. Dieser Mittelwert wird für jede Fläche errechnet und dann sortiert.

So, die Theorie hätten wir schon hinter uns gelassen, die Praxis sieht kaum anders aus. Das GFA-BASIC-Programm besteht aus einem Objekteditor, einem Darstellungsmodus sowie etlichen Funktionen, die einem das dreidimensionale Konstruieren erleichtern.



```
SN = SIN(Neigung),      CN=  COS(Neigung)
SR = SIN(Richtung),     CR=  COS(Richtung)
SK = SIN(Kippung),      CK=  COS(Kippung)              (5)
```

```
XTURN = (CK*CR+SN*SR*SK) * X + (SN*SR*CK-SK*CR)  * Y + (SR*CN) * Z
YTURN = (SK*CN)          * X + (CK*CN)           * Y + (-SN)   * Z
ZTURN = (SK*CR*SN-SR*CK) * X + (CK*CR*SN+SK*SR)  * Y + (CR*CN) * Z   (6)
```

Programmpraxis

Da die Grundlagen und die Programmlänge eh schon die Grenzen eines Artikels erreichen, muß auf eine ausführliche Programmbeschreibung verzichtet werden. Interessierte Leser können sich anhand der Programmkommentierung durchhangeln. Die nun folgende, eh wichtigere Programmbedienung dürfte auch noch einiges klären.

Das Listing läuft in allen drei Auflösungsstufen, wobei die mittelere nicht zu empfehlen ist, da die Objekte wegen des Pixelverhältnisses recht verzogen aussehen.

Nach dem Programmstart erscheint rechts eine Menüleiste, die mit Mausklick (links und rechts!) bedient werden kann. Links davon ist das pixelquadratische Eingabe- und Anzeigefeld (bei mittlerer Auflösung leider arg unquadratisch und klein).

In den ersten drei Menüpunkten steht die X-,Y-, Z-Position, wobei X und Y durch die Positionierung der Maus im Eingabefeld bestimmt werden. Der Z-Wert gibt die jeweilige Tiefe an und kann durch Mausdruck am rechten oder linken Rand des Menüs verändert werden. Bei einem Klick in die Mitte des Menüpunkts kann der Z-Wert auch numerisch über die Tastatur eingegeben werden.

Die Erstellung eines Objekts beginnt mit einem Linksklick auf das Menü "PUNKTMOD", wonach im Eingabefenster mit Linksklick Punkte gesetzt werden können. Mit Rechtsklick wird der nächstliegende Punkt gelöscht. Die aktuelle Punktposition ist in der X-,Y-,Z-Angabe abzulesen. Eine weitere Möglichkeit ist die numerische Punkteingabe durch Rechtsklick auf "PUNKTMOD|XYZ-EING". Die Punkte werden nach der Entfernung vom Betrachter in drei unterschiedlichen Größen dargestellt.

Nach der Eingabe möglichst aller Punkte schaltet man durch einen Klick auf "FLÄCHEN ERSTELLEN" in den Flächenmodus. Nun sollte eine Fläche nach der Korkenzieherregel durch Anklicken der einzelnen Punkte eingegeben werden. Am besten wählt man die einem am nächsten liegende Fläche und klickt die Punkte mit der linken Maustaste im Uhrzeigersinn an. Der jeweils zuletzt geklickte Punkt wird durch einen Kreis gekennzeichnet, die Teillinien werden durchgezogen dargestellt. Sollte man einen falschen Punkt erwischt haben, genügt ein Rechtsklick, und die letzte Verbindung wird gelöscht. Als letzten Punkt der Fläche wählt man den Ausgangspunkt und nun zeigt einem das Programm eine Auswahlbox mit den wählbaren Farben und Mustern. Ein Klick auf das Gewünschte reicht, und die Fläche ist vollständig definiert. Sie wird dann mit gepunkteten Linien dargestellt.

Nach der Eingabe der ersten Fläche sollten die daran anliegenden Flächen eingegeben werden. Nur eine Fläche, die mit einer bereits bestehenden über eine gemeinsame Linie verfügt, kann vom Programm auf 'Korkenzieherrichtigkeit' geprüft werden (Sage noch einer Informatik sei trocken!). Nach der Eingabe weiterer Flächen werden die Linien, die mehr als eine Fläche berühren, gestrichelt dargestellt. Dies vereinfacht die Konstruktion ausgefüllter Objekte sehr, da dann nur noch die fehlenden oder gepunkteten Linien zu Flächen zusammengefaßt werden müssen. Ein vollständiges Objekt besteht lediglich aus gestrichelten Linien. Leider frißt die Berechnung gemeinsamer Linien soviel Rechenzeit, daß diese Hilfe über den Menüpunkt "INFO + ANZEIGEMODUS" umschaltbar ist. Im Modus 'Schnell' werden alle Linien einfach gepunktet dargestellt.

Objektdemo

Sobald einige Flächen erstellt sind oder gar das ganze Objekt fertig ist (als Übungsobjekte eignen sich besonders schiefe Dreieckspyramiden und Würfel!), kann "3D-GRAFIK STARTEN" aufgerufen werden und, siehe da - unser Objekt überschlägt sich in voller Größe und Schönheit. Durch Mausbewegung kann der Standpunkt des Objekt verändert werden. Mit einem Rechtsklick gelangt man in das Hauptmenü zurück, mit einem Linksklick bleibt das Objekt stehen und kann in aller Ruhe betrachtet werden. Jetzt sind auch gezielte Veränderungen der Werte über die Tastatur möglich. So läßt sich das Objekt mit "+" und "-" zoomen, durch "0" in die Ausgangslage zurückbringen, und mit den Tasten "n", "r" und "k" lassen sich die Drehwinkel bestimmen. Jeder andere Tastendruck zeigt eine kleine Hilfestellung.

So, eine noch unerwähnte Funktion ist "GITTER | SETZEN", die sich umschalten und im Pixelmaß bestimmen läßt. Dieses Gitter hilft bei der Punkteingabe, indem jeder Wert in X- und Y-Position auf die Ecken des Gitters begrenzt wird, was vielleicht auch als SNAP-Funktion bekannt ist.

Eigene Erweiterungen sind problemlos machbar, am einfachsten durch Abändern der DATA-Einträge "FREI" und Einbinden des Prozedurnamens in den Befehlsaufruf "On Com Gosub...". Ich empfehle auch die Verwendung der Prozeduren *Inbox(Info$,Anzahl[1 oder 3])* und *Ali(Parameter wie bei Alert)*, wodurch man sich einige Mühe sparen kann.

Die Menüeinträge "RESERVIERT" sind für die Erweiterungen im nächsten Artikel gedacht, sollten also nicht für eigene Prozeduren verwendet werden. Und damit sich niemand Kopf und Finger an Problemen zerbricht, die die nächste ST eh schon löst, eine kleine Vorschau auf die "RESERVIERT"-Prozeduren:

Einstellung der Tiefenverzerrung, Load & Save, Flächeneditor zum Umfärben und Löschen von Flächen sowie ein Menüpunkt, der einen ganzen Quader auf einmal erstellt.

Martin A. Wielebinski

```
 1:  '
 2:  ' ***********************************************
 3:  ' *
 4:  ' * 3D-Grafik Editier- und Demoprogramm für die ST
       v. Martin A. Wielebinski.
 5:  ' *
 6:  ' ***********************************************
 7:  '
 8:  On Error Gosub Fehler ! Kann nie schaden
 9:  Deftext 1,0,0,4       ! Immer Minitext
10:  Xs=640                ! totale x Breite 640
11:  Ys=400                ! totale y Höhe 400
12:  Tyo=14                ! Y-Offset für die Menütexte
13:  If Xbios(4)<>2        ! nicht hochauflösend ?
14:    Ys=200              ! ja, y Höhe 200
15:    Tyo=7               ! neuer Y-Textoffset
16:    If Xbios(4)=0       ! niedrigste Auflösung ?
17:      Xs=320            ! ja, x Breite 320
18:    Endif
19:  Endif
20:  Xmi=Xs Div 2          ! X-Mitte
21:  Ymi=Ys Div 2          ! Y-Mitte
22:  Xsl=Xs-1              ! Letzter wählbarer X-Wert
23:  Ysl=Ys-1              ! dito Y
24:  Xtx=Ys+(Xs-Ys)/2-8    ! X-Offset für die Menütexte
25:  Yty=Ys/20             ! Y-Texthöhe des Menüs
26:  '
27:  Disk$="\"             ! Default Pfad
```

Auto-Monitor-Switchbox

Neu:
Mit automatischer Erkennung der Auflösung. Das Starten von Programmen in der falschen Auflösung gehört der Vergangenheit an. Programm anklicken: Programm wird automatisch in der richtigen Auflösung (Farbe oder Monochrom) gestartet.

Softwaremäßiges Umschalten o. Reset erstmals möglich **TASTATURRESET,** und Umschalten über die Tastatur.

Auto-Monitor-Switchbox

in neuer Version



Funktion Switchbox

Mit der **AUTO Monitor Switchbox** können Sie über die Tastatur aus jedem Programm heraus zwischen Monochrom und Farbmonitor umschalten oder einen Tastaurreset durchführen. Desweiteren ist es möglich über die Tastatur einen Kaltstart durchzuführen. Die mitgelieferte Software ist resetfest (arbeitet ständig im Hintergrund). Desweiteren ist es möglich durch Einbinden der von uns mitgelieferten Routinen **ohne RESET** zwischen Monochrom und Farbe umschalten.

Die Software wird nun in einer neuen Version ausgeliefert, die es ermöglicht, daß das Programm automatisch in der richtigen Auflösung gestartet wird. Das ständige Starten von Programmen in der falschen Auflösung gehört der Vergangenheit an.

Die AUTO Monitor-Switchbox verfügt zusätzlich über einen BAS und Audio Ausgang. Die Verarbeitungsqualität wird auch Sie überzeugen.

Update Software:
gegen Einsendung der Original Diskette + 15.00 DM.

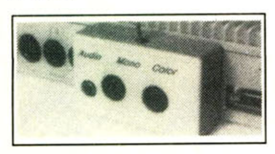
Monitor Switchbox

Monitor Switchbox, die sich nur über einen mechanischen Schalter zwischen Monochrom und Farbe umschalten läßt.

Monitorswitchbox mit 45 cm Kabel, zusätzlicher BAS und Audio Ausgang **44.90**

Monitor Switchbox direkt an den Rechner anstöpselbar mit zusätzlichem Audioausgang **39.90**

dto. als Bausatz (komplett) **29.90**



Switchbox direkt an den Rechner anstöpselbar. (Alle Ausgänge sind weiterhin frei erreichbar, (s. Abb.) Jedoch nur für die Modelle 260/520/1040 ST ohne eingebauten Modulator geeignet.

Modul A.-M.-Switchbox

Mit diesem Modul stehen Ihnen alle Funktionen der AUTO Monitor Switchbox zu Verfügung.
Desweiteren verfügt dieses Modul über mehrere wichtige Tools die das Arbeiten mit dem Computer erleichtern, z.B. Fast Load, Mousebeschleuniger, Druckerspooler etc.
Das **Modul AUTO Monitorswitchbox** wird einfach in den ROM Port Ihres Rechners gesteckt. Die Garantie Ihres Computers bleibt erhalten. **59.90**

Audiokabel Monitor Switchbox 5 polig, ca. 2 m **7.90**

Audiokabel Monitor Switchbox Cynch, ca. 2 m **9.90**

Preise

Auto Monitor Switchbox
mit zusätzlichen BAS u. Audio Ausgang 45 cm Monitorkabel incl. Software **59.90**

Auto Monitor Switchbox ST
direkt an den Rechner anstöpselbar mit zusätzlichem Audio Ausgang incl. Software **54.90**
Wichtig: nur für die Rechnertypen 260/520/1040 ST ohne eingebauten HF Modulator geeignet.

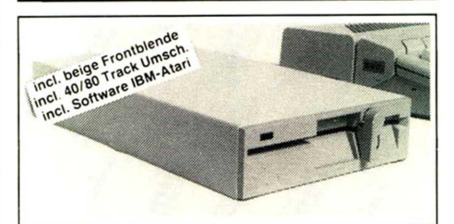
Auto Monitor Switchbox Multisync
incl. Verbindungskabel Multisyncmonitor zusätzlicher Audio Ausgang, incl. Software
Wichtig: Monitortyp angeben **69.90**

Diskettenlaufwerke

3,5-Zoll und 5,25-Zoll-Diskettenlaufwerke in vollendeter Qualität. Es werden nur die besten Materialien verwendet, z.B. Netzteil 5,25 Laufwerk vom größten europäischen Netzteilhersteller mit VDE, GS, Thermosicherung. Die Laufwerksgehäuse sind mit einer kratzfesten Speziallackierung ausgestattet. Die Chassis werden **nicht** über den Direktimport bezogen. Dieses ermöglicht eine ständige Lieferbereitschaft und einen guten Service.

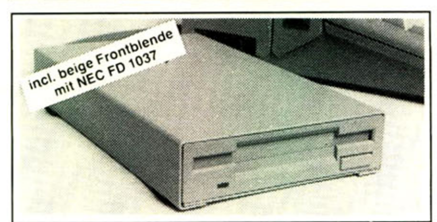
Qualität und Service, der auch Sie überzeugen wird.

5,25 Disketten-Laufwerk



Anschlußfertiges 5,25 Zoll Diskettenlaufwerk (720 KB)
Laufwerkstyp **TEAC FD 55 FR,** robuste Verarbeitung. Unterstützt PC DITTO,
Aufpreis 2. Floppybuchse 27.90, Schaltung A/B 20.- **339.-**

3,5 Disketten-Laufwerk



Anschlußfertiges 3,5-Zoll-Diskettenlaufwerk (720 KB)
Laufwerkstyp: **NEC FD 1037**
(25,4 mm hoch) robuste Verarbeitung, kratzfestes Gehäuse Netzteil VDE,
Einbau 2. Floppbuchse 27.90 **249.-**

Software a la carte

Omikron Compiler	159.-	**Signum II**	357.-
		STAD	159.-

PC DITTO Version 3.64 **169.-**

Public-domain (alle ST Disketten)
incl. Diskette eins. format.
DM 5,90
Liste gegen DM 2,- in Briefmarken

PD Pakete 1 Paket = 5 Disk (doppelseitig) **35.-**

- P.1 **Die besten Spielprogramme**
- P.2 **Die besten Utilitieprogramme**
- P.3 **Die besten Mal/Soundprogramme**
- P.4 **Die besten diversen Programme**

Hardware Zubehör

Leerkarte Speichererweiterung
komplett bestückte Speichererweiterung (steckbar) ohne RAM's **84.90**

Epromkarte
128 KB **49.90**

Leerkarte Speichererweiterung
4 MB (Computertyp angeben) **249.-**

Epromkarte 512 KB
incl. Treibersoftware zum Programmieren der Eproms **119.-**

HF Modulator HF Modulator zum Anschluß des Atari ST an jeden gewöhnlichen Farbfernseher. Der HF Modulator zeichnet sich durch die besonders gute Bildwiedergabe aus. Der Ton wird über den Fernseher übertragen.
incl. FBAS, AUDIO Ausgang, Attenenkabel, hochwertiges Netzteil **189.-**

Festplatten

Neu von Hard & Soft
professionelle Festplatten für den Atari ST.
- AUTO-Boot-fähig
- Durchgeführter DMA-Port
- schneller als die Atari Festplatte
- Festplattenkabel standardmäßig 2 m lang
- sehr leise (kein störender Lüfter)
- Abmessungen wie die orig. Atari Festplatte (gleichzeitig als Monitoruntersatz verwendbar)
- sehr robuste Verarbeitung — 1 Jahr Garantie

30 MB nur 1198.-
60 MB nur 1798.-

Fordern Sie unser kostenloses Zusatzinfo an.

abgesetzte Tastatur am ST

Abgesetzte Tastatur am ST
Tastaturgehäuse mit Spiralkabel, Treiberstufe Resettaste und Joystickportbuchsen eingebaut **124.-**

ST Tast (XT Tastatur am ST)
Jede XT Tastatur am ST anschließbar, frei definierbare Tastaturbelegung, Tastaturreset **139.-**

ST Tast + orig. Cherry Tastatur
Kombiangebot **329.-**

Tastaturabdeckgehäuse Das Tastaturabdeckgehäuse wird einfach über den Atari ST gestülpt. Alle Ausgänge des Atari sind frei erreichbar. Durch diesen Zusatz wird der Computer zum idealen Monitorständer. Ideale Ergänzung zum Tastaturgehäuse und ST Tast **59.90**

Hardware Zubehör

Uhrmodul intern gegenüber anderen Uhrmodulen benötigen Sie keine Software zum Anerkennen der Uhr. Die Bootsoftware befindet sich auf ROM's im Betriebssystem.
Wichtig: Betriebssystem angeben. ROM TOS o. Blitter TOS **119.-**

Uhrmodul extern
incl. Treibersoftware **89.-**

Mouse Pad rutschfeste Unterlage für Computermäuse (270 x 220 mm) **17.90**

3 Laufwerke am ST

Floppyswitchbox: ermöglicht den Anschluß von drei Laufwerken am ST. Das Umschalten erfolgt ohne Reset. Die Switchbox ist mit speziellen Treibern für 3,5 u. 5,25 Laufwerke ausgestattet (m. Zugriffs-LED-Anzeige)
Wichtig: Computertyp angeben **89.-**

Multisync Monitore

Monitor NEC Multisync GS — die echte Alternative zum SM 124
Auflösung 900 x 700, alle drei Auflösungen des Atari in SW darstellbar, bestechend scharfe Bildqualität, gebaut nach ergonomischen Richtlinien mit integriertem Schwenk-/Neigefuß
1 Jahr Garantie (keine Importware) nur **579.-**

Monitor NEC Multisync II Auflösung: 800 x 560, alle drei Auflösungen vom Atari ST darstellbar, bestechend scharfe Bildwiedergabe, gebaut nach ergonomischen Richtlinien mit integrietem Schwenk-/Neigfuß
1 Jahr Garantie (keine Importware) nur **1698.-**

Verbindungskabel

Floppyverlängerung (0,7 m)	29.90
Monitorverlängerung (1,5 m)	29.90
Harddiskkabel (1,5 m)	39.90
Druckerkabel (2 m)	24.90
Scartkabel (2 m)	39.90

Stecker

Monitorstecker	6.40
Monitorbuchse	8.90
Monitorkupplung	9.90
Floppystecker	7.90
Floppybuchse	9.90
Floppykupplung	9.90

Disketten

3,5 Fuji 2 DD	34.90
3,5 Fuji 2 DD (neutral verpackt)	27.90
5,25 Precision 2 DD (5,25 Zoll Disketten höchster Qualität)	10.90

Zubehör/Disketten-LW

Floppykabel Atari 3,5 Disketten-LW	27.90
Floppykabel Atari 5,25 Disketten-LW	29.90
Floppygehäuse NEC FD 1036, 1037 Teac FD 55 FR, mit Zugentlastung u. Gummifüße	29.-
NEC FD 1037 mit Anschlußbelegung	197.-
Teac FD 55 FR 229.- Atari modif.	239.-

Hard & Soft A. Herberg, Bahnhofstr. 289, 4620 Castrop-Rauxel, Tel. 0 23 05/1 57 64, Fax. 12022 — Händleranfragen erwünscht —

```
29:  Darsty=11              ! und Y
30:  Gittab=10              ! Default Gitterabstand =10
31:  Grmbo=1                ! Intelligente, aber
                            ! langsame Darstellung
32:  '
33:  Pun_max=299 ! Anzahl der maximal möglichen Punkte
34:  Fl_max=199  ! Anzahl der maximal möglichen Flächen
35:  Eck_max=19  ! Maximale Eckenzahl einer Fläche
36:  '
37:  Dim Sk%(Fl_max) ! Nummernfeld für die Sortierung
                     der Flächen
38:  Dim Zh%(Fl_max) ! Speicher für die Tiefe der
                     Flächenmitten
39:  Dim Xyz%(2,Pun_max) ! Das Feld für die Punkt-
                         koordinaten
40:  Dim Xy%(2,Pun_max)  ! Feld der umgerechnete Punkte
41:  Dim Fl%(Eck_max+3,Fl_max) ! Das Feld für die
                               Flächennum.und Infos
42:  Dim Xyz0%(2)   ! Standpunkt des Betrachters
43:  Dim Nrk(2)     ! Neigung, Richtung, Kippung
44:  Dim Arbxyz%(2) ! Arbeitsnullpunkt (für die Eingabe)
45:  Dim X%(Eck_max),Y%(Eck_max)  ! 2 Felder für die
                                  Polyfillfunktion
46:  '
47:  Color 4
48:  For I=0 To 19 ! Bildschirm initialisieren
49:    Box Ys,Yty*I,Xsl,Yty*(I+1)  ! Box ausgeben
50:    Read A$
51:    Text Ys+(Xs-Ys)/2-Len(A$)*3,Yty*I+Tyo,A$
                                ! und  den Menütext
52:  Next I
53:  Line Ys,Ysl,Xs,Ysl
54:  Get Ys,0,Xsl,Ysl,Iscr$ ! Menüleiste abspeichern
55:  '
56:  Gosub Show                 ! Koordinatenkreuz
57:  Defmouse 5                 ! Fadenkreuz
58:  Do                         ! und ab in die
                                  Hauptschleife:
59:    Mouse Mx,My,Mk
60:    If Mx<Ys And (Mx<>Omx Or My<>Omy) ! Maus im
                                Bild und neuer Wert ?
61:      Text Xtx,Tyo,"       "  ! ja, Werte anzeigen...
62:      Text Xtx,Yty+Tyo,"       "
63:      Text Xtx,Tyo,Arbxyz%(0)-Ymi+Mx        ! x +
64:      Text Xtx,Yty+Tyo,Arbxyz%(1)+Ymi-My
                                     !y ausgeben und
65:      Omx=Mx
66:      Omy=My           ! als bekannte Werte übernehmen
67:    Endif
68:    If Mk              ! Maustaste gedrückt ?
69:      If Mx>Ys         !  ja, in Menüleiste ?
70:        Com=(My Div Yty)+1 ! ja, Befehl errechnen
                              und ausführen:
71:        If Com<10
72:          On Com GosubD,D,Wert,In_gr,D,Gitt,Phol,
             Fhol,G3d
73:        Else
74:          On Com-9 Gosub D,D,D,D,D,D,D,D,D,D,D
75:        Endif
76:        Defmouse 5
77:        Gosub Show ! Bildschirmausgabe erneuern.
78:      Else         ! falls nicht in der Menüleiste:
79:        On Punbo Gosub P_akt ! Im Punktmodus
                              Punkt setzen/löschen
80:        On Flaebo Gosub F_akt ! Im Flächenmodus
                                setzen/rücksetzen
81:      Endif
82:    Endif
83:  Loop              ! Auf ewig...
84:  '
85:  Procedure In_gr   ! Gibt Informationen aus und
                       schaltet den Ausgabemodus um
86:    Local A$
87:    A$="Das Objekt hat "+Str$(P_anz)+" Punkte,|und
                       "+Str$
       (F_anz)+" Flächen.|"
88:    Gosub Ali(2,A$+"Grafikmoduswählen:",0," Schnell|
                                              Schön")
89:    Grmbo=But-1
90:  Return
91:  '
92:  Procedure Xyz_show
93:    Local I
94:    For I=0 To 2 ! Arbeitsnullpunkt anzeigen...
95:      Text Xtx,Tyo+Yty*I,"       "
96:      Text Xtx,Tyo+Yty*I,Arbxyz%(I)
97:    Next I
98:  Return
99:  '
100:  Procedure Wert ! Arbeitsnullpunkt versetzen
101:    Local A$
102:    Pause 8        ! für günstige Bedienung
103:    If Mx<Ys+30  ! "-" gedrückt ?
104:      Repeat
105:        Dec Arbxyz%(Com-1) ! Ja erniedrigen bis
                               Taste losgelassen
106:        Gosub Xyz_show
107:      Until Mousek=0
108:    Else
109:      If Mx>Xs-30                ! "+" ?
110:        Repeat
111:          Inc Arbxyz%(Com-1)     ! dito erhöhen
112:          Gosub Xyz_show
113:        Until Mousek=0
114:      Else ! ansonsten numerische Werteingabe
115:        A$="BITTE DEN NEUEN|STANDARTWERT FÜR
            DIE|"+Chr$(88+Com-1)
116:        Gosub Inbox(A$+" KOORDINATE EINGEBEN.",1)
117:        Arbxyz%(Com-1)=Wert
118:      Endif
119:    Endif
120:  Return
121:  '
122:  Procedure Gitt ! Gitter schalten und Maßstab eing.
123:    Local A$
124:    If Mk=1        ! Umschalten ?
125:      A$="Gitter einschalten ???|Es vereinfacht die
                          |Eingabe von Punkten."
126:      Gosub Ali(2,A$,Gittbo+1,"An|Aus")
127:      If But=Gittbo+1
128:        Gittbo=2-But
129:      Endif
130:    Else          ! Ansonsten Maßstab entgegennehmen
131:      Gosub Inbox("GITTERABSTAND SETZEN.|BISHERIGER
            ABSTAND: "+Str$(Gittab),1)
132:      If Wert
133:        Gittab=Wert
134:      Endif
135:    Endif
136:  Return
137:  '
138:  Procedure Phol ! Punktmodus umschalten oder
                     XYZ-Eingabe
139:    Local I,A$
140:    If Mk=2       ! XYZ-Koordinateneingabe ?
141:      Do          ! ja:
142:        A$="Einen Punkt anhand|der XYZ-Koordinaten|
            eingeben ?????"
143:        Gosub Ali(2,A$,2," JA |Nein")
144:        Exit If But=2
145:        A$="FÜR DEN NEUEN PUNKT|JETZT BITTE DIE|"
146:        Gosub Inbox(A$+" KOORDINATEN EINGEBEN.",3)
147:        Gosub P_neu(X%(0),X%(1),X%(2))
148:      Loop        ! Noch einen Punkt ???
149:    Else
150:      A$="Punktmodus einschalten ?|Links Punkte
                        setzen,|Rechts löschen."
151:      Gosub Ali(2,A$,Punbo+1," JA |Nein")
152:      If But=Punbo+1
153:        Punbo=2-But
154:        Flaebo=0
155:      Endif  ! ansonsten Punktmodus umschalten
156:    Endif
157:  Return
158:  '
159:  Procedure Fhol ! Flächenmodus ein/ausschalten
160:    Local A$
161:    A$="Flächenerstellung starten ?|Einfach mit der
        Maus|Punkte verbinden."
162:    Gosub Ali(2,A$,Flaebo+1," JA |Nein")
163:    If But=Flaebo+1
164:      Flaebo=2-But
165:      Punbo=0
166:    Endif
167:  Return
168:  '
169:  Procedure P_akt ! Maus-Punkteingabe bzw.Löschakt.
170:    Local X,Y,I
171:    X=Arbxyz%(0)-Ymi+Mx
172:    Y=Ymi-Arbxyz%(1)-My      ! Punktdaten umrechnen
173:    If Mk=2 And P_anz>0      ! Punkt löschen ???
174:      Gosub Punget(X,Y)      ! ja, also suchen
```

Listing geht weiter...

```
175:        Xyz%(0,Nr)=Xyz%(0,P_anz)
176:        Xyz%(1,Nr)=Xyz%(1,P_anz)
177:        Xyz%(2,Nr)=Xyz%(2,P_anz)    ! löschen durch
                                          umsetzen
178:        Dec P_anz                   ! Punktanzahl -= 1
179:      Else                  ! ansonsten Punkteingabe:
180:        If Gittbo ! falls Gitter vorhanden ->anpassen
181:          X=Int((X+Gittab/2)/Gittab)*Gittab
182:          Y=Int((Y+Gittab/2)/Gittab)*Gittab
183:        Endif
184:        Gosub P_neu(X,Y,Arbxyz%(2))     ! und Punkt
                                            eintragen
185:      Endif
186:      Gosub Show                ! Veränderungen anzeigen
187:      Gosub Mclr
188:    Return
189:    '
190:    Procedure P_neu(X,Y,Z) ! Neuen Punkt aufnehmen
191:      Local I,Bo
192:      Bo=1                  ! Im Zweifel neuer Punkt
193:      If P_anz
194:        For I=1 To P_anz ! Alle Punkte durchgehen
195:          If Xyz%(0,I)=X And Xyz%(1,I)=Y And
              Xyz%(2,I)=Z
196:            Bo=0            ! falls bereits vorhanden
197:            Pnr=I           ! und Pnr gleich die Nummer
198:          Endif
199:        Next I
200:      Endif
201:      If Bo                 ! falls Punkt noch nicht
                                  vorhanden
202:        Inc P_anz           ! Punktzähler += 1
203:        Xyz%(0,P_anz)=X     ! und Punkt eintragen
204:        Xyz%(1,P_anz)=Y
205:        Xyz%(2,P_anz)=Z
206:        Pnr=P_anz           ! Pnr = Punktnummer
207:      Endif
208:    Return
209:    '
210:    Procedure Punget(X,Y)   ! Diese Procedure sucht
                                  den Punkt, der am
211:      Local I,Ab,Wzae       ! nächsten bei X,Y liegt.
212:      Ab=2000               ! Default Abstand
213:      Nr=0                  ! Default Punktnummer
214:      For I=1 To P_anz      ! Alle Punkte durchgehen
215:        Wzae=Abs(Xyz%(0,I)+(Xyz%(2,I)-Arbxyz%(2))/
            Darstx-X)
216:        Wzae=Wzae+Abs(Xyz%(1,I)+(Xyz%(2,I)-Arb
            xyz%(2))/
              Darsty-Y)
217:        If Wzae<Ab          ! Falls neuer Abstand
                                  kleiner als bisher
218:          Ab=Wzae           ! neuen Abstand
219:          Nr=I              ! und Punktnummer merken
220:        Endif
221:      Next I
222:    Return
223:    '
224:    Procedure F_akt         ! Flächenerstellung:
225:      Local I,J,K,X,Y,P1,P2,Turn,Pp1,Pp2
226:      If P_anz>2            ! Genügend Punkte für
                                  eine Fläche vorhanden?
227:        X=Arbxyz%(0)-Ymi+Mx
228:        Y=Ymi-Arbxyz%(1)-My   ! x,y umrechnen
229:        Gosub Punget(X,Y)     ! Punkt suchen
230:        If Mk=1               ! neuen Punkt setzen ???
231:          If Flnr>0           ! ja, neue Fläche ?
232:            J=1               ! nein
233:            For I=0 To Flnr   ! neuer Punkt bereits
                                    vorhanden ?
234:              If Fl%(I,F_anz)=Nr
235:                J=0           ! jawoll, also Quark!
                                    (oder Fläche fertig)
236:              Endif
237:            Next I
238:          Endif
239:          If J=1 Or Flnr=0    ! Erster Punkt oder kein
                                    Quark ?
240:            If Flnr=Eck_max
241:              Gosub Ali(1,"Soviele Ecken|sind nicht
                                            |  möglich
                    !!!",1,"Honk!")
242:            Else
243:              Fl%(Flnr,F_anz)=Nr! ja, abspeichern
244:              Inc Flnr              ! Flächennummerzähler
                                          erhöhen
```

```
245:              Fl%(Eck_max+1,F_anz)=Flnr-1! und abspei
246:            Endif
247:          Endif
248:          If Fl%(0,F_anz)=Nr And Flnr>2 ! Fläche fertig
249:            Gosub Colget        ! Farbe oder Muster holen
250:            Fl%(Eck_max+2,F_anz)=Cnr     ! abspeichern
251:            Fl%(Eck_max+3,F_anz)=Czei
252:            Turn=0
253:            If F_anz     ! Bei anliegenden Flächen auf
254:              For I=0 To F_anz-1 ! Korkenzieherregel
                                       prüfen
255:                For J=1 To Fl%(Eck_max+1,I)+1
256:                  P1=Fl%(J-1,I) ! Mit den alten
                                      Linien vergleichen
257:                  If J=Fl%(Eck_max+1,I)+1
258:                    P2=Fl%(0,I)
259:                  Else
260:                    P2=Fl%(J,I)
261:                  Endif
262:                  For K=1 To Flnr
263:                    Pp1=Fl%(K-1,F_anz)
264:                    If K=Flnr
265:                      Pp2=Fl%(0,F_anz)
266:                    Else
267:                      Pp2=Fl%(K,F_anz)
268:                    Endif
269:                    If P1=Pp1 And P2=Pp2
                                  !Bei gleicher Ausrichtung
270:                      Turn=1! ist die Regel verletzt
271:                    Endif
272:                  Next K
273:                Next J
274:              Next I
275:            Endif
276:            If Turn             ! Regel verletzt
277:              For I=0 To Flnr-1  ! Dann Feld
                                       umsortieren
278:                X%(Flnr-1-I)=Fl%(I,F_anz)
279:              Next I
280:              For I=0 To Flnr-1
281:                Fl%(I,F_anz)=X%(I)
282:              Next I
283:            Endif
284:            Inc F_anz   ! Anzahl der Flächen erhöhen
285:            Flnr=0 ! Flächenecken zähler wieder auf 0
286:          Endif
287:        Else      ! ansonsten letzen Punkt löschen
288:          If Flnr>0      ! ist noch einer da ?
289:            Dec Flnr     ! jawoll, vergiß ihn.
290:          Endif
291:        Endif
292:        Gosub Show
293:      Endif
294:    Return
295:    '
296:    Procedure Colget ! Hier werden Farbe und Muster
                           zur Auswahl angeboten
297:      Local I,J,Zw$
298:      Get Xmi-100,Ymi-40,Xmi+100,Ymi+40,Zw$
                                          ! Platz schaffen
299:      Deffill 0
300:      Pbox Xmi-100,Ymi-40,Xmi+100,Ymi+40
301:      Color 1
302:      Box Xmi-100,Ymi-40,Xmi+100,Ymi+40 ! Eingabebox
303:      Holcol:
304:      For J=0 To 2
305:        For I=0 To 15
306:          If J=0
307:            Deffill I,2,8
308:          Else
309:            Deffill 1,J+1,I+1
310:          Endif
311:          Pbox Xmi+10*(I-8),Ymi+20*(J-1.5),Xmi+10*
                   (I-8)+9,Ymi+20*(J-1.5)+19
312:        Next I
313:      Next J              ! alle Möglichkeiten anbieten
314:      Gosub Mhol          ! Mausklick abwarten
315:      Cnr=(Mx-(Xmi-80)) Div 10
316:      Czei=(My-(Ymi-30)) Div 20     ! Wahl errechnen
317:      If Cnr<0 Or Cnr>15 Or Czei<0 Or Czei>2
318:        Goto Holcol       ! War Käse, als o von vorne
319:      Endif
320:      If Czei<>0          ! Musternummer anpassen
321:        Inc Cnr
322:        Inc Czei
323:      Endif
```

```
324:    Put Xmi-100,Ymi-40,Zw$ ! Bildschirm restaurieren
325:  Return
326:  '
327:  Procedure Show      ! Phänomenale Anzeigeroutine
328:    Local I,J,Ii,Jj,X,Y,X2,Y2,P1,P2,Pp1,Pp2 ! alle
        r Ereignisse
329:    Deffill 0
330:    Pbox 0,0,Ysl,Ysl    ! Bild löschen
331:    Color 8
332:    If Gittbo          ! Wenn nötig Gitter anzeigen
333:      For I=-((Arbxyz%(0)-Ymi) Mod Gittab)
          To Ysl Step Gittab
334:        Line I,0,I,Ysl
335:      Next I
336:      For I=((Arbxyz%(1)+Ymi) Mod Gittab) To Ysl
          Step Gittab
337:        Line 0,I,Ysl,I
338:      Next I
339:    Endif
340:    Color 2
341:    If Abs(Arbxyz%(0))<Ymi   ! Falls möglich
                                 Koordinatenkreuz
342:      Line Ymi-Arbxyz%(0),0,Ymi-Arbxyz%(0),Ysl
343:    Endif
344:    If Abs(Arbxyz%(1))<Ymi                ! ausgeben
345:      Line 0,Ymi+Arbxyz%(1),Ysl,Ymi+Arbxyz%(1)
346:    Endif
347:    '
348:    If F_anz     ! Die Linien aller Flächen anzeigen
349:      For J=0 To F_anz-1
350:        For I=1 To Fl%(Eck_max+1,J)+1
351:          P1=Fl%(I-1,J)
352:          If Fl%(Eck_max+1,J)+1=I
353:            P2=Fl%(0,J)   ! Der letzte Punkt ist der
                               erste
354:          Else
355:            P2=Fl%(I,J)
356:          Endif
357:          Defline 3
358:          If Grmbo          ! Sortiert ?
359:            Jj=0
360:            Do
361:              If J<>Jj
362:                Ii=1
363:                Do
364:                  Pp1=Fl%(Ii-1,Jj)
365:                  If Fl%(Eck_max+1,Jj)+1=Ii
366:                    Pp2=Fl%(0,Jj)    ! Der letzte
                                   Punkt ist der erste
367:                  Else
368:                    Pp2=Fl%(Ii,Jj)
369:                  Endif
370:                  If P1=Pp2 And P2=Pp1
371:                    Defline 2    ! Mehrfache als
                                     Strichlinien
372:                    Jj=F_anz
373:                    Ii=Eck_max+1 ! Untersuchung
                                       fertig
374:                  Endif
375:                  Inc Ii
376:                  Exit If Ii>Fl%(Eck_max+1,Jj)+1
377:                Loop
378:              Endif
379:              Inc Jj
380:              Exit If Jj>=F_anz
381:            Loop
382:          Endif
383:          X=Ymi-Arbxyz%(0)+Xyz%(0,P1)+(Xyz%(2,P1)-
                Arbxyz%(2))/Darstx
384:          Y=Ymi-Arbxyz%(1)-(Xyz%(1,P1)+(Xyz%(2,P1)-
                Arbxyz%(2))/Darsty)
385:          X2=Ymi-Arbxyz%(0)+Xyz%(0,P2)+(Xyz%(2,P2)-
                Arbxyz%(2))/Darstx
386:          Y2=Ymi-Arbxyz%(1)-(Xyz%(1,P2)+(Xyz%(2,P2)-
                Arbxyz%(2))/Darsty)
387:          Line X,Y,X2,Y2
388:        Next I
389:      Next J
390:    Endif
391:    '
392:    Defline 1
393:    If Flnr>1 ! Die Linien einer neuen Fläche ausge.
394:      For I=1 To Flnr-1    ! Durchzählen
395:        P1=Fl%(I-1,F_anz)
396:        P2=Fl%(I,F_anz)
397:          X=Ymi-Arbxyz%(0)+Xyz%(0,P1)+(Xyz%(2,P1)-
                Arbxyz%(2))/Darstx
398:          Y=Ymi-Arbxyz%(1)-(Xyz%(1,P1)+(Xyz%(2,P1)-
                Arbxyz%(2))/Darsty)
399:          X2=Ymi-Arbxyz%(0)+Xyz%(0,P2)+(Xyz%(2,P2)-
                 Arbxyz%(2))/Darstx
400:          Y2=Ymi-Arbxyz%(1)-(Xyz%(1,P2)+(Xyz%(2,P2)-
                 Arbxyz%(2))/Darsty)
401:        Line X,Y,X2,Y2
402:      Next I
403:    Endif
404:    '
405:    If P_anz                    ! Punkte vorhanden ?
406:      Color 1
407:      For I=1 To P_anz          ! ja, alle ausgeben
408:        X=Ymi-Arbxyz%(0)+(Xyz%(0,I)+(Xyz%(2,I)-
              Arbxyz%(2))/Darstx)
409:        Y=Ymi-Arbxyz%(1)-(Xyz%(1,I)+(Xyz%(2,I)-
              Arbxyz%(2))/Darsty)
410:        If Xyz%(2,I)>Arbxyz%(2)
411:          Draw X-1,Y To X,Y-1 To X+1,Y To X,Y+1 To
              X-1,Y
412:        Else
413:          If Xyz%(2,I)=Arbxyz%(2)! Je nach Z-Wert
                                   unterschiedlich
                                   groß
414:            Line X-1,Y-1,X+1,Y+1
415:            Line X-1,Y+1,X+1,Y-1
416:          Else
417:            Line X-2,Y-2,X+2,Y+2
418:            Line X-2,Y+2,X+2,Y-2
419:          Endif
420:        Endif
421:        If Flnr>0
422:          If I=Fl%(Flnr-1,F_anz)
423:            Circle X,Y,4
424:          Endif
425:        Endif
426:      Next I
427:    Endif
428:    Put Ys,0,Iscr$
429:    Gosub Xyz_show              ! Nullpunkt ausgeben
430:    Color Punbo
431:    Circle Ys+5,6.5*Yty,2     ! Punktmodus anzeigen
432:    Color Flaebo
433:    Circle Ys+5,7.5*Yty,2   ! Flächenmodus anzeigen
434:  Return
435:  '
436:  Procedure G3d                 ! Grafikshow
437:    Local I,Mkk,T$,Scr$,Osp,Nsp
438:    If F_anz>0
439:      Scr$="Mit der linken Maustaste|können Sie
            die Ansicht|verändern, mit der"
440:      Gosub Ali(0,Scr$+" rechten|die Grafik wieder
            verlassen.",1,"Klasse!")
441:      Cls
442:      Xyz0%(2)=500      ! Z-Startwert
443:      Osp=Xbios(2)
444:      Scr$=Space$(32512)
445:      Nsp=Varptr(Scr$) ! Einen zweiten Bildschirm
446:      Nsp=((Nsp+256) Div 256)*256
447:      Gosub Mclr
448:      Hidem
449:      Karneval:
450:      Repeat
451:        Void Xbios(5,L:Nsp,L:Osp,-1)
452:        Gosub Verdreht
453:        Void Xbios(5,L:Osp,L:Nsp,-1)
454:        Gosub Verdreht
455:        Mkk=Mousek
456:      Until Mkk
457:      Void Xbios(5,L:Osp,L:Osp,-1)
458:      If Mkk=1
459:        Gosub Mclr
460:        Repeat
461:          T$=Inkey$
462:          If T$<>""
463:            Gosub Befehle
464:            Gosub Transform
465:            Gosub Anzeige
466:            If T$="?"
467:              Print "  Befehle ZOOM: '+', '-';
                  NRK = 0: '0';";
468:              Print "   'N'eigung, 'R'ichtung,
                  'K'ippung."
469:            Endif
```

Listing geht weiter..

```
470:          Endif
471:        Until Mousek
472:        Gosub Mclr
473:        Goto Karneval
474:      Endif
475:      Showm
476:      Gosub Mclr
477:    Endif
478:  Return
479:  '
480:  Procedure Verdreht
481:    For I=0 To 2
482:      Nrk(I)=Nrk(I)+0.2/(I+1)
483:    Next I              ! Winkel weiterbewegen
484:    Gosub Transform     ! Transformation
485:    Gosub Anzeige       ! Bildschirmausgabe
486:    Xyz0%(0)=Mousex-Xmi
487:    Xyz0%(1)=Ymi-Mousey
488:  Return
489:  '
490:  Procedure Befehle
491:    If T$>="a" And T$<="z"  ! Nur Großbuchstaben
492:      T$=Chr$(Asc(T$)-32)
493:    Endif
494:    If T$="+"
495:      Sub Xyz0%(2),16
496:    Endif
497:    If T$="-"
498:      Add Xyz0%(2),16
499:    Endif
500:    If T$="0"
501:      For I=0 To 2
502:        Nrk(I)=0
503:      Next I
504:    Endif
505:    If T$="N" Or T$="R" Or T$="K"
506:      Gosub Inbox("BITTE GEBEN SIE DEN WERT|FÜR
      "+T$+" IN ALTGRAD AN.",1)
507:      Wert=Wert/180*Pi
508:    Endif
509:    If T$="N"
510:      Nrk(0)=Wert
511:    Endif
512:    If T$="R"
513:      Nrk(1)=Wert
514:    Endif
515:    If T$="K"
516:      Nrk(2)=Wert
517:    Endif
518:    If Instr("+-NRK0",T$)=0
519:      T$="?"
520:    Endif
521:  Return
522:  '
523:  Procedure Anzeige          ! Anzeige des Objekts
524:    Local I,F,Ii,Ll,P1,P2,P3,Sk,Xx,Yy,Zz
525:    For I=1 To P_anz        ! Alle Punkte
526:      P1=Xyz%(0,I)
527:      P2=Xyz%(1,I)
528:      P3=Xyz%(2,I)
529:      Xx=Tm1*P1+Tm2*P2+Tm3*P3+Xyz0%(0) ! Punkte
                                          verdrehen
530:      Yy=Tm4*P1+Tm5*P2+Tm6*P3+Xyz0%(1) ! und zum
                                          Standort
531:      Zz=Tm7*P1+Tm8*P2+Tm9*P3+Xyz0%(2) ! tranfor-
                                          mieren
532:      Xy%(0,I)=Xmi+Xx*512/Zz ! Perspektivverkürzung
533:      Xy%(1,I)=Ymi-Yy*512/Zz ! und Y auf den Kopf
                                 stellen,
534:      Xy%(2,I)=Zz          ! alles zwischenspeichern.
535:    Next I
536:    For F=0 To F_anz-1    ! Alle Flächen durchgehen
537:      Ll=Fl%(Eck_max+1,F)
538:      Zh%(F)=0
539:      For I=0 To Ll
540:        Zh%(F)=Zh%(F)+Xy%(2,Fl%(I,F))
541:      Next I
542:      Zh%(F)=Zh%(F)/Ll    ! Der Z-Mittelwert bilden
543:      Sk%(F)=F
544:      If F>0
545:        For I=F Downto 1
546:          If Zh%(I-1)<Zh%(I)         ! und nach Z
                                          sortieren
547:            Swap Zh%(I-1),Zh%(I)
548:            Swap Sk%(I-1),Sk%(I)
549:          Endif
550:        Next I
551:      Endif
552:    Next F
553:    Cls
554:    For Ii=0 To F_anz-1    ! Nochmals alle Flächen
555:      F=Sk%(Ii)            ! Sortiert ausgeben
556:      P1=Fl%(0,F)    ! Die ersten drei Punkte nehmen
557:      P2=Fl%(1,F)
558:      P3=Fl%(2,F)
559:      Sk=(Xy%(0,P2)-Xy%(0,P1))*(Xy%(1,P3)-
          Xy%(1,P2))
560:      Sk=(Xy%(0,P3)-Xy%(0,P2))*(Xy%(1,P2)-
          Xy%(1,P1))-Sk
561:      If Sk<0                     ! Sichtbar ?
562:        P1=Fl%(Eck_max+3,F)
563:        P2=Fl%(Eck_max+2,F)
564:        If P1                     ! ja, Füllfarbe
565:          Deffill 1,P1,P2
566:        Else
567:          Deffill P2,1
568:        Endif
569:        Ll=Fl%(Eck_max+1,F) ! Anzahl der Eckpunkte
570:        For I=0 To Ll
571:          X%(I)=Xy%(0,Fl%(I,F))
572:          Y%(I)=Xy%(1,Fl%(I,F)) ! Für Polyfill
                                   umsetzen
573:        Next I
574:        X%(Ll+1)=X%(0)
575:        Y%(Ll+1)=Y%(0)
576:        Polyfill Ll+1,X%(),Y%()  ! Malen
577:      Endif
578:    Next Ii                      ! alle Flächen...
579:  Return
580:  '
581:  Procedure Transform      ! Die Neigung, Richtung,
                             Kippung vorberechnen
582:    Local Sn,Cn,Sr,Cr,Sk,Ck
583:    Sn=Sin(Nrk(0))
584:    Cn=Cos(Nrk(0))
585:    Sr=Sin(Nrk(1))
586:    Cr=Cos(Nrk(1))
587:    Sk=Sin(Nrk(2))
588:    Ck=Cos(Nrk(2))           ! Damit es schneller geht
589:    Tm1=Ck*Cr+Sn*Sr*Sk
590:    Tm2=Sn*Sr*Ck-Sk*Cr
591:    Tm3=Sr*Cn
592:    Tm4=Sk*Cn
593:    Tm5=Ck*Cn
594:    Tm6=-Sn
595:    Tm7=Sk*Cr*Sn-Sr*Ck
596:    Tm8=Ck*Cr*Sn+Sk*Sr
597:    Tm9=Cr*Cn                ! Die Transformation
598:  Return
599:  '
600:  Procedure Inbox(Bla$,Anr)  ! Einfache Eingabebox
601:    Local I,Y,Zw$
602:    Get Xmi-Xmi/2,Ymi-Ymi/2,Xsl,Ymi+Ymi/2,Zw$
603:    Deffill 0
604:    Pbox Xmi-Xmi/2,Ymi-Ymi/2,Xmi+Xmi/2,Ymi+Ymi/2
605:    Color 5
606:    Box Xmi-Xmi/2,Ymi-Ymi/2,Xmi+Xmi/2,Ymi+Ymi/2
607:    Color 4
608:    Pbox Xmi-Xmi/2+2,Ymi-Ymi/2+2,Xmi+Xmi/2-
    2,Ymi+Ymi/2-2
609:    Pbox Xmi-Xmi/2+3,Ymi-Ymi/2+3,Xmi+Xmi/2-
    3,Ymi+Ymi/2-3             ! ein bißchen Schönheit
610:    Y=Ymi-Ymi/3
611:    If Anr=1
612:      Bla$=Bla$+"||EINGABE MIT|<RETURN>BESTÄTIGEN"
613:    Else
614:      Bla$=Bla$+"||EINGABEFORMAT:|X,Y,Z <RETURN>"
615:    Endif
616:    Do
617:      I=Instr(Bla$,"|")
618:      Exit If I=0
619:      Text Xmi-3*(I-1),Y,Left$(Bla$,I-1)
      ! Den Text zentriert untereinander
620:      Bla$=Right$(Bla$,Len(Bla$)-I)    ! ausgeben
621:      Add Y,10
622:    Loop
623:    Text Xmi-3*Len(Bla$),Y,Bla$ ! Noch den Resttext
624:    Print At(Xs/320*12,16); ! Cursor setzen
625:    If Anr=1           ! Eingabe von einem Wert
626:      Input Wert$      ! und ein  primitives Input
627:      Wert=Val(Wert$)  ! für numerische Werte
628:    Else
```

```
629:       Input X%(0),X%(1),X%(2)  ! Eingabe von drei
                                       Werten
630:     Endif
631:     Put Xmi-Xmi/2,Ymi-Ymi/2,Zw$ ! aufräumen
632:     Gosub Mclr
633:   Return
634:   '
635:   Procedure Ali(Ic,Inf$,Bunr,Ant$) ! Alertbox ohne
                                          Quietschen
636:     Repeat
637:     Until Mousek=0 And Inkey$=""
638:     Alert Ic,Inf$,Bunr,Ant$,But
639:     Defmouse 5                 ! und Mausveränderung.
640:   Return
641:   '
642:   Procedure D                  ! (D)ummy
643:     Gosub Ali(1,"Dieser Befehl existiert nicht!
                  ",1,"Na klar!")
644:   Return
645:   '
646:   Procedure Mclr ! Wartet bis keine Maustaste
                          klickt
647:     Repeat
648:     Until Mousek=0
649:   Return
650:   '
651:   Procedure Mhol       ! Entprellte Maustasteneingabe
652:     Gosub Mclr
653:     Repeat
654:       Mouse Mx,My,Mk
655:     Until Mk
656:     Gosub Mclr
657:   Return
658:   '
659:   Procedure Fehler     ! Fehler abfangen...
660:     Local A$
661:     On Error Gosub Fehler
662:     A$="Das ist leider ein|Fehler Nr.:
          "+Str$(Err)+" aufgetreten."
663:     Gosub Ali(3,A$,1,"Huch!")
664:     Resume Next
665:   Return
666:   '
667:   Data "X:                  "
668:   Data "Y:                  "
669:   Data "- Z:               +"
670:   Data INFO + ANZEIGEMODUS
671:   Data RESERVIERT
672:   Data GITTER | SETZEN
673:   Data PUNKTMOD|XYZ-EING
674:   Data FLÄCHEN ERSTELLEN
675:   Data 3D-GRAFIK STARTEN
676:   Data RESERVIERT
677:   Data RESERVIERT
678:   Data FREI,FREI,FREI
679:   Data FREI,FREI,FREI
680:   Data RESERVIERT,RESERVIERT,RESERVIERT
```

Listing: 3D-Grafik Edier-und Demoprogramm

ENDE

Multilinguale Textverarbeitung

mit Signum2! und Sigrevers

In den Geisteswissenschaften spricht sich allmählich herum, daß der Computer die tägliche Arbeit erleichtern kann. Zu diesem Zweck kaufte ich mir, als Student der ev. Theologie, vor 1 1/2 Jahren einen ATARI 1040 ST und das Textverarbeitungsprogramm SIGNUM. Dieses Programm beinhaltet unter anderem einen sehr einfach zu bedienenden Fonteditor (Font = Zeichensatz), der mich dazu reizte, einen speziellen Hebräischfont für unsere Arbeit am Lehrstuhl für Altes Testament (Prof. Dr.D. Michel) zu erstellen. Ein Neutestamentler regte mich daraufhin an, einen analogen Font für neutestamentliche Texte (Griechisch) zu erstellen, da er seine Dissertation damit erfassen wollte (inzwischen ist die auf einem NEC P6 mit Carbonbändern ausgedruckte Vorlage für den Offset vom Verlag angenommen worden). Um die entstandenen Fonts einem weiteren Kreis von Interessenten zukommen zu lassen, übernahm Application Systems /// Heidelberg die Fonts als *SIFOX Theologie* in ihren Pool von SIFOX-Fonts (SIFOX=Signum Font Exchange) auf.

Informationsknotenpunkt

Auf einen kleinen Artikel in der Zeitschrift "computer persönlich", in dem ich über unsere Arbeit berichtete, meldeten sich Interessenten aus dem gesamten Bundesgebiet. Es stellte sich aber heraus, daß das System noch nicht 100% in der multilingualen Textverarbeitung eingesetzt werden konnte. Es fehlte ein Zusatzmodul, das das Rechts-links-Schreiben ermöglicht.

SIGREVERS das komfortable Rechts-links-Schreiben

Die bei mir eingegangenen Wünsche wurden an die Firma Application Systems weitergeleitet und blieben nicht unbeantwortet. Zuerst erschien Mitte 1987 die erweiterte Version SIGNUM 2!, die Accessories zuläßt.

Anfang März 1988 durfte ich dann das erste Accessory testen. SIGREVERS ermöglicht den SIGNUM-Anwendern nicht bloß das Schreiben in Rechts-links-Richtung, sondern es erlaubt auch, den Zeilenumbruch am linken Rand vorzunehmen und die einzelnen Zeichen direkt zu überschreiben (s. Bild 1).

ARABICUM - THEOLFONT plus

Da uns, die wir Hebräisch schreiben, nun ein mächtiges Arbeitszeug an die Hand gegeben war, wuchs der Anwenderkreis stetig an. An unserem Fachbereich Ev.Theologie, der Uni Mainz, hatte sich der ATARI bereits als mächtiges Werkzeug etabliert, aber auch andere Universitäten wurden auf unsere Arbeit aufmerksam. Damit die noch fehlenden Fonts erstellt werden konnten, erklärten sich Professoren und wiss. Mitarbeiter mehrerer Universitäten zur Mithilfe bereit.

Die vorhandenen SIFOX-Theologie-Fonts wurden komplett überarbeitet, durch Zusatzfonts erweitert (z.B. bieten die Griechischzusatzfonts nun fast alle Akzentkombinationen) und im Fußno-

Arabisch:

قال ٱلنَّبِىُّ صلَّى الله عليه وسلَّم : صُحْبَةُ ٱلْعاقِلِ زِيادَةٌ فى ٱلدِّينِ وصُحْبَةُ ٱلْجاهِلِ نُقْصَانٌ فى ٱلدِّينِ

وقال أَيْضًا : اَلْمُؤْمِنُ لا يَنْجُو مِن عَذَابِ ٱللّٰهِ تَعَالَى حتى يَتْرُكَ أَرْبَعَةَ أَشْيَاءٍ ٱلْكِذْبَ

Griechisch:

1 Ἐν ἀρχῇ ἦν ὁ λόγος, καὶ ὁ λόγος ἦν πρὸς τὸν θεόν, καὶ θεὸς ἦν ὁ λόγος. 2 οὗτος ἦν ἐν ἀρχῇ πρὸς τὸν θεόν. 3 πάντα δι᾽ αὐτοῦ ἐγένετο, καὶ χωρὶς αὐτοῦ ἐγένετο ⸀οὐδὲ ἕν⸁. ὃ γέγονεν 4 ἐν αὐτῷ ζωὴ ⸂ἦν, καὶ ἡ ζωὴ ἦν τὸ φῶς ⸋τῶν ἀνθρώπων⸌· 5 καὶ τὸ φῶς ἐν τῇ σκοτίᾳ φαίνει, καὶ ἡ σκοτία αὐτὸ οὐ κατέλαβεν.

Hebräisch:

1 1 בְּרֵאשִׁ֖ית בָּרָ֣א אֱלֹהִ֑ים אֵ֥ת הַשָּׁמַ֖יִם וְאֵ֥ת הָאָֽרֶץ׃ 2 וְהָאָ֗רֶץ הָיְתָ֥ה תֹ֙הוּ֙ וָבֹ֔הוּ וְחֹ֖שֶׁךְ עַל־פְּנֵ֣י תְה֑וֹם וְר֣וּחַ אֱלֹהִ֔ים מְרַחֶ֖פֶת עַל־פְּנֵ֥י הַמָּֽיִם׃ 3 וַיֹּ֥אמֶר אֱלֹהִ֖ים יְהִ֣י א֑וֹר וַֽיְהִי־אֽוֹר׃

tenformat gezeichnet. Herr Blum, der Lehrbeauftragter für Jiddisch an der Uni München ist, regte einen speziellen Hebräischfont für das Jiddische an. Zur Abrundung wurde die neu entstandene Fontdiskette *THEOLFONT plus* noch um einen Koptischfont erweitert.
Auf der Fontdiskette *ARABICUM* befinden sich 4 Arabisch-Persischfonts (2 Größen - 2 Zusatzfonts, u.a. mit Ligaturen) sowie 4 umfassende Umschriftfonts (gerade, kursiv, und in der Schriftart TIMES in 2 Größen). Beide Disketten können direkt bei der Firma Application Systems, über Ihren Händler oder bei mir bezogen werden.

Hebräische Wortkunde

Eines unserer Teilprojekte, die Wortkunde des biblischen Hebräisch, gab den Anstoß zur Schaffung der Fontdisketten *ARABICUM* und *THEOLFONT* plus. Die Wortkunde wird von Herrn Prof. Konrad Ehlich (Linguist - Uni Dortmund) verfaßt und von Herrn Prof. Dr. Diethelm Michel (Altes Testament - Uni Mainz) als Herausgeber betreut.

Aufbau

Die neuartige Wortkunde soll das Erlernen der hebräischen Vokabeln erleichtern. Der erste Teil führt in die Benutzung der Wortkunde ein und wird von einer kurzen Lehre der hebräischen Nominalbildung abgeschlossen. Den Hauptteil bildet der Wortteil, der insgesamt etwa 1000 Vokabeln aufführt, von denen 500 wegen ihres häufigeren Vorkommens und der größeren Bedeutung im AT vorrangig gelernt werden sollten. Hierzu werden diese Vokabeln nebst den Übersetzungshilfen und Erläuterungen in einer größeren Schrifttype erfaßt (s.Bild 2).
Der Wortteil ist in 3 Spalten unterteilt. In der 1. Spalte steht das hebräische Wort und in der 2. eine erste Grundübersetzung. Die 3. Spalte bietet weitere Bedeutungen (z. B. עָוֹן awōn - *Vergehen, Schuld* und *Strafe*). Durch Pfeile (—>, <—) werden dabei Entwicklungen innerhalb dieser semantischen Struktur, des Bedeutungsbereiches, angegeben. Die Abgrenzung dieser Bedeutungsbereiche der einzelnen Wörter geschieht in der Sprache vor allem durch benachbarte und entgegengesetzte Begriffe. Die Wörter, die untereinander so gegenseitig bezogen sind, sich gegenseitig bestimmen, formen ein sprachliches Spannungsfeld, ein ⬭ - *Wortfeld*. Daneben finden sich z.B. noch Hinweise auf typische Wendungen, Nominalbildungsklassen und Lehnwörter aus dem Jiddischen.

Fontbedarf

Die Zweiteilung der 1000 Vokabeln bedingt, daß alle Fonts, die in der Wortkunde vorkommen, je in einem 11er- (11 Punkt Größe) und 9er- (9 Punkt Größe) Format vorliegen müssen. Daraus ergäbe sich rein rechnerisch ein Bedarf von 10 Fonts, da neben Deutsch und Hebräisch noch Griechisch, Umschrift und spezielle Editionszeichen benötigt werden. Signum gestattet aber *nur*, 7 Fonts gleichzeitig in einem Dokument zu verwenden.
Dabei kommt uns der Umstand zu Hilfe, daß den Umschriftfonts die ohnehin verwendeten TIMES-Fonts zugrunde liegen. Deshalb können die kleinen Buchstaben des Alphabetes hier weggelassen werden. Es bleiben also nur wenige Zeichen übrig, die benötigt werden. Ebenso bedarf es nur relativ weniger Editionszeichen (z.B. das Zeichen für *Wortfeld* ⬭ und ○ - wie Sie sehen, können auch die Anführungszeichen nach Typensatzregeln geändert werden), so daß 4 Fonts in einen zusammengefaßt werden können. Am Schluß blieb sogar noch Platz für einige Ligaturen übrig, um das Hebräische noch korrekter wiederzugeben (—> Font *WORDKADD*).

Problemlösungen

Wie Sie aus den Beispielen ersehen können, steht in der 3. Spalte oft ein Text im Blocksatz. Dabei gehe ich folgendermaßen vor: Zuerst wurden 3 Tabulatorpositionen festgelegt - je zum Beginn einer

fem. אֹזֶן	Ohr	dual אָזְנַיִם ('ŏznájim) NB qutl
אזן	hi. הֶאֱזִין hinhören	zuhören; mit acc. auf etwas hören denominiert ⬭ שמע, הִקְשִׁיב
אַחַר אַחֲרֵי	hinter (örtl.) nach (zeitl.)	אַחַר als Adverb: hinten
חנן	gnädig sein	← jmdm. geneigt sein hitp. הִתְחַנֵּן um Gnade flehen (← sich Gnade zu verschaffen suchen) ⬭ חמל, רחם, הֵיטִיב wohl Basisverwandtschaft mit חנה [Johannes ← Ἰωάννης ← יוֹחָנָן = „Jahwe ist gnädig"]
חֶסֶד	Gemeinschaftspflicht	Loyalität Ausdruck des Verhältnisses, in dem man zu anderen steht (Verwandtschaft, Freundschaft, Gastfreundschaft, Zugehörigkeit, Dienstverhältnis) עשׂה חֶסֶד Loyalität erweisen חֶסֶד וֶאֱמֶת עִם wirklichen Häsäd (Loyalität als Erfüllung der Verpflichtung zu ihm) NB qatl

Einige Beispiele aus der „Wordkunde des biblischen Hebräisch"

1 2 3 4 5 6 7 8 9 0 ß ´ # ( ) / *
ESC … BACK SPACE
Q W E R T Z U I O P Ü + 7 8 9 -
TAB â à â à é ē é ē ū ū í ī ō ŏ ō ŏ DEL
A S D F G H J K L Ö Ä - 4 5 6 +
CONTROL š ṣ š ṣ ḫ ḥ í ī RETURN
« Y X C V B N M , . - 1 2 3
SHIFT á ā á ā SHIFT ENTER
ALT CAPS LOCK 0 .

Fontbeispiel WORDKADD

Spalte. Wenn in der 3. Spalte dann ein längerer Text zu erwarten ist, der später in Block gesetzt werden soll, springe ich mit der *Tab*-Taste zur 3. Spalte, drücke Esc e (einrücken markieren) und betätige danach die *Return*-Taste. Nachdem der Text der 3. Spalte geschrieben ist, wird er in Block gesetzt (dies kann natürlich auch nachträglich vorgenommen werden). Nach Positionierung des Cursor in der ersten Zeile (1. und 2. Spalte) wird die erste Zeile des Blockes mit *Ctrl F7* hochgezogen. Jetzt positioniert man auf das erste Wort der 3. Spalte und verschiebt mit *Ctrl Tab* den Zeilenrest bis zur nächsten Tabulatorposition, womit die 3. Spalte fertig wäre. Es hört sich komplizierter an, als es in der Praxis ist.

Daneben bietet SIGNUM 2! die willkommene Hilfe der Tastenprogrammierung. Diese ist einfach zu bewerkstelligen und erleichtert die Arbeit ungemein. Über *F1* "-" (F = Funktionstaste) löse ich z.B. folgende Arbeitsschritte aus: *Esc a* (Absatz), Ziehen der durchgehenden Linie, Return, Esc a (Absatz). Zusätzlich habe ich noch kompliziertere Makros erstellt, die in den Fußnotenbereich und die zu schreibende Zeile positionieren, den Zeilenabstand umstellen, die Fußnotenfonts auf die Tastatur legen etc. - alles nur über *F1 und betreffende Taste auszulösen.*
Um Speicherplatz zu sparen, ignoriert SIGNUM die Leerzeichen, was nur beim automatischen Unterstreichen stört. In der Wortkunde werden aber alle deutschen Übersetzungen unterstrichen, so daß die herkömmliche Methode des Unterstreichens mehrerer Wörter lästig wäre. Um Abhilfe zu schaffen, habe ich auf 3 Tasten des Zusatzfonts Unterstriche in 3 verschiedenen Längen gelegt, die dort zu stehen kommen, wo auch der automatische Unterstrich steht. Wenn ich also mehrere Wörter zu unterstreichen habe, wird automatisches Unterstreichen mit Esc u eingeschaltet, und zwischen den Wörtern drücke ich statt der Leertaste die gewünschte Taste aus dem Zusatzfont.

SIGREVERS im Dauerbetrieb

Aus den Beispielen kann leicht ersehen werden, wie oft zwischen der Links-rechts- und der Rechts-links-Schreibrichtung gewechselt werden muß. Bis heute trat kein Problem mit SIGREVERS auf. Zu schätzen weiß ich die Möglichkeit, mit F9 direkt zu überschreiben, da es sich stets um vokalisiertes Hebräisch handelt. Gegenüber den Systemen auf einem IBM-Kompatiblen (z.B. mit WordPerfect) haben wir den Vorteil, alles so auf dem Monitor zu sehen, wie es später ausgedruckt wird, während es z.B. bei unseren Kollegen, die mit WordPerfect arbeiten, ständig zu Frust kommt, weil sie die Vokalzeichen (über Steuerzeichen eingegeben - verschwinden nach Eingabe, wenn in Normalschrift umgeschaltet wird) erst am Ausdruck kontrollieren und korrigieren können.

Zukünftige Anwendungen

Für unsere zukünftigen Anwendungen wünschen wir uns noch eine erweiterte ASCII-Tabelle zum Einlesen von Daten (z.B. Hebräisch) aus einem Datenbankprogramm (bei uns ADIMENS) in SIGNUM. Laut Auskunft des SIGNUM-Programmierers dürfen wir damit in der nächsten Programmversion rechnen. Andere Wünsche werden hoffentlich in näherer oder fernerer Zukunft realisiert werden können, als da wären Registererstellung, Datenbanksystem mit den Fähigkeiten von SIGNUM (z.B. für Wörterbuch und Konkordanzerstellung).

Syrisch

Neben Hebräisch, Griechisch, Koptisch und Arabisch benötigen wir Theologen, aber nicht nur wir, noch das Syrische. In einer Gemeinschaftsproduktion Oxford (England)-Mainz-München werden in der nächsten Zeit die beiden wichtigen Syrischdialekte Serto und Estrangela entstehen (nach dem Schriftbild der Oxford University Press). Nach Fertigstellung werden sie wohl den registrierten Anwendern der *THEOLFONT plus* gegen einen geringen Unkostenbeitrag zur Verfügung gestellt werden.

Tagung in Bethel

Vom 10.-13. Oktober nahm ich an der Tagung *Computer und Wissenschaftliche Theologie* in der Kirchlichen Hochschule Bethel teil. Als offizieller Vertreter des Fachbereiches Ev. Theologie der Uni Mainz sollte ich unsere Computeranwendungen vorstellen. Es zeigte sich, daß wir mit ATARI-Computern in Verbindung mit SIGNUM 2! und den *THEOLFONTS* das beste Textverarbeitungssystem für Griechische und Hebräische Anwendungen haben (z.B. im Vergleich zu *WORD + LOGOS* oder *WordPerfect*). Diesen Vorteil erkaufen wir uns z.Z. leider noch mit dem Mangel an gewissen Zusatzangeboten, die *WORD* oder *WordPerfect* haben (z.B. automatische Registererstellung). Wenn man aber den Bedienungskomfort von SIGNUM und SIGREVERS berücksichtigt, fällt die Entscheidung auf jeden Fall zu unseren Gunsten aus. Leider - daran konnte unsere Demonstration auch nichts ändern - wollen viele DOS-Fans die objektiven Tatsachen einfach nicht zur Kenntnis nehmen. Ein interessantes Zukunftsprojekt könnte die Kombination EUMEL-ATARI-SIGNUM bringen. Vom Betriebssystem Eumel und besonderer Zusatzsoftware unterstützt soll die *Dialogfähige Computerkonkordanz* (AT) zur wissenschaftlichen Analyse eingesetzt werden können. Da der ATARI im Systemfont den kompletten Konsonantenbestand des Hebräischen bietet, wird er als Terminal eingesetzt. Durch SIGNUM sollen dann die gefundenen Verse weiterbearbeitet (z.B. vokalisiert) und in SIGNUM-Qualität ausgedruckt werden können.

Anwenderkreis

Über meine Fontdisketten habe ich inzwischen gelernt, wie weitgefächert der Interessentenkreis ist. Neben Professoren, Doktoren, Promoventen und Studenten der altsprachlichen Fächer sowie Pfarrern sind dies *interessierte Laien*. Er reicht von der Bibliothekarin über die pensionierte Musikwissenschaftlerin bis zum Tierarzt, der sich als Numismatiker auf den Orient spezialisiert hat. Seit kurzem gibt es die Fontdisketten THEOLFONT plus und ARABICUM auch für Frankreich und England (gleicher Zeichenbestand aber andere Belegung).

Als Theologen bedrückt mich aber die Zahl der schwarzen Schafe, die nicht registrierte Anwender sind. Ich werde damit konfrontiert, wenn ich auf eine Bestellung hin, zuerst einmal nach der Registrierungsnummer frage. Meistens erhalte ich dann keine Antwort mehr. Hoffentlich merken diese Anwender bald, daß sie den Ast absägen, auf dem sie sitzen. Gute Software kann nur dann noch verbessert werden, wenn sie weiterhin gekauft und nicht schwarz kopiert wird.
Wer sich näher informieren will und den frankierten Rückumschlag nicht vergißt (1,30 DM), darf sich an die folgende Adresse wenden:

Veit Brixius
Römerstr. 48
6501 Budenheim
Tel.: 06139/6504

ENDE

ST COMPUTER

ABO

ST COMPUTER

ABO

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57

ST COMPUTER

Einzelheft- u. Monatsdisketten Bestellung

ST COMPUTER

Einzelheft- u. Disketten Service

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57

ST COMPUTER

Kleinanzeigen

ST COMPUTER

Kleinanzeigen

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57



Abonnement

Ja, bitte senden Sie mir die ATARI-Computer Fachzeitschrift ab ______________________
für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 70,– frei Haus.
(Ausland: Nur gegen Scheck-Voreinsendung DM 90,– Normalpost. DM 120,– Luftpost)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Name

Vorname

Straße/Nr.

PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM______ liegt bei.

☐ Vorauskasse per Post-Einzahlung (Zahlkarte)

Garantie:
Diese Bestellung kann ich schriftlich innerhalb einer Woche (rechtzeitige Absendung genügt) widerrufen. Dies bestätige ich durch meine 2. Unterschrift.

Datum Unterschrift

Datum Unterschrift



ABO



Einzelheft- u. Monatsdisketten Bestellung

ST-Computer können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 6,— (1987) DM 7,— (1988) nachbestellen. Bearbeitung nur gegen beigefügten Scheck über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli	Aug./Sept.	Okt.	Nov.	Dez.

1987 = DM
1988 = DM

+ Gebühr für Porto und Verpackung = DM

(1 Heft DM 2,–, ab 2 Hefte DM 5,–)

☐ Scheck in Höhe zus. DM liegt bei

Disketten Service

Alle Programme, die in ST-Computer veröffentlicht wurden, sind auf Disketten erhältlich. Die Disketten enthalten die Programme von jeweils 2 ST-Computer-Ausgaben. Bestellen Sie durch ankreuzen die gewünschten Disketten (* Monate mit Stern DM 18,—)

* Preis je Diskette 28,— DM	Jan./Febr.	März/Apr.	Mai/Juni	Juli/Aug.*	Sept./Okt.	Nov./Dez.	Jan./Febr.	März/Apr.	Mai/Juni
	87	87	87	87	87	87	88	88	88
	Juli*	Aug./Sept./Okt.	Nov./Dez.	Jan./Febr.	März/Apr.	Mai/Juni	Juli/Aug.	Sept./Okt.	Nov./Dez.
	88	88	88	89	89	89	89	89	89

Lieferung: gegen beigefügten Scheck zuzügl. 5,— DM Versandkosten, unabhängig der bestellten Menge.



Einzelheft- u. Monatsdisketten Bestellung



Kleinanzeigen-Auftrag

Bitte veröffentlichen Sie für mich folgende Kleinanzeige in der angekreuzten Rubrik

Biete an ☐ Hardware ☐ Software
Ich suche ☐ Hardware ☐ Software
☐ **Tausch** ☐ **Kontakte**
☐ Verschiedenes ☐

30 Buchstaben je Standardzeile – incl. Satzzeichen und Wortzwischenräume.
Groß- und Kleinbuchstaben verwenden, fettgedruckte Wörter unterstreichen.

Bearbeitung **nur gegen Vorausscheck** über den entsprechenden Betrag (keine Überweisung)

☐ privat = DM 7,– je Zeile
☐ gewerblich = DM 15,– je Zeile
☐ Chiffregebühr = DM 10,–

☐ Scheck über DM__________ ist beigefügt

Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

Datum Unterschrift

Absenderangaben auf der Rückseite nicht vergessen

Kleinanzeigen

ST COMPUTER

Kontaktkarte

ST COMPUTER

Kontaktkarte

Bitte Adresse der Firma, bei der Sie Informationen, oder etwas bestellen möchten, auf der rechten Seite eintragen → → → → → → → → → →

Absender
(Bitte deutlich schreiben)

Vorname/Name

Beruf

Straße/Nr.

PLZ/Ort

Telefon Vorwahl/Rufnummer

Bitte freimachen

Postkarte

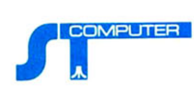
Firma

Straße/Postfach

PLZ Ort



Kurzmitteilung



Kurzmitteilung



Postkarte

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Telefon

MAXON - Computer GmbH
ST-Computer Redaktion
Industriestraße 26

6236 Eschborn



PD Bestellung



PD Bestellung

Bitte freimachen

Postkarte

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

MAXON Computer GmbH
ST-Computer Redaktion
Industriestraße 26

6236 Eschborn

ST COMPUTER

☐ Ich bitte um weitere Informationen

☐ Ich gebe folgende Bestellung auf

in Bezug auf Ihre Anzeige in ST-Computer Heft ______ Seite ___

Menge	Produkt/Bestellnummer	DM	gesamt DM

Datum, Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)

Abgesandt am:

ST COMPUTER

Firma:

Bemerkungen:

ST COMPUTER

Meine Meinung

Zu dem Artikel ______________ in Heft ____, Seite ____
hätte ich folgendes zu bemerken:

☐ Ich möchte Ihnen folgendes Programm zur Veröffentlichung anbieten: (Kurzbeschreibung, Sprache, Länge in Druckerseiten, GEM/TOS)

☐ Ich kann über folgendes Thema berichten: (Tips & Tricks am ST, Hardware, Software, etc.)

☐ Ich möchte gerne Autor in der ST-Computer werden. Meine Fachgebiete: (z.B. LISP, Pearl, Modula-2, Assembler ...)

☐ Ich möchte, daß folgendes Public-Domain Programm in Ihre Sammlung aufgenommen wird.

☐ Sonstiges

Bei weiteren Angaben oder Fragen wenden Sie sich bitte schriftlich oder telefonisch an die Redaktion. Tel. 0 61 96/48 18 11

ST COMPUTER

Kurzmitteilung

ST COMPUTER

PUBLIC DOMAIN SERVICE

Ich bestelle folgende PD-Disketten:
(Siehe PD Service in dieser Ausgabe)

Zahlung erfolgt:

☐ per Scheck

☐ per Nachnahme

Je Diskette fügen Sie bitte einen Betrag von DM 10,– bei,
für Porto und Verpackung je Sendung DM 5,– (Ausland DM 10,–)

Datum

Unterschrift

ST COMPUTER

PD Bestellung

Datum

Suchen Sie einen Partner, der Sie in der Schweiz kompetent vertritt?

Sie wollen Ihre Software Atari-Fachhändlern in der Schweiz anbieten und suchen eine kompetente und zuverlässige Vertretung? Wir von DataTrade in Zürich sind ein junges, dynamisches Team, welches bereits gut 60 namhafte Software-Hersteller vertritt und über 270 Händler beliefert.

Wir würden uns auch sehr gerne für Sie und Ihr Produkt voll einsetzen. Möchten Sie uns kennenlernen? So nehmen Sie doch unverbindlich mit unserem Geschäftsführer, Herrn Eric Hofmann Kontakt auf, schriftlich oder telefonisch.
Wir freuen uns auf eine künftige Zusammenarbeit!



DTZ DataTrade AG · Langstrasse 94 · Postfach · CH - 8021 Zürich · Telefon 01/242 80 88 · Telefax 01/291 05 07

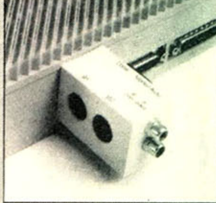
Das aktuelle Angebot von GALACTIC

HF / Video-Modulatoren

Wir bieten Geräte für jedes Anwendungsgebiet an:

MOD2	das Standardmodell mit Fernseheranschluß	**DM 170,-**
MOD3	mit integrierter Monitorumschaltbox	**DM 195,-**
MOD3 auto	zusätzlich Automonfähig (s.u.)	**DM 205,-**
MOD3a	Video- (FBAS)Ausgang mit Umschaltbox	**DM 150,-**
MOD3a auto	ebenfalls Automonfähig	**DM 160,-**
Für Preisbewußte: (solange Vorrat reicht):		
MOD2a	Videogerät ohne Umschaltbox	nur **DM 120,-**
Sonderanfertigungen auf Anfrage		
Passendes Netzteil		**DM 10,-**

Die etwas andere Umschaltbox



Unsere Umschaltbox U2 kann nicht nur wie andere zum manuellen oder automatischen Wechsel zweier verschiedener Monitore eingesetzt werden, zusätzlich dienst sie als Monitorverteiler zweier gleichartiger Bildschirme (RGB + RGB oder Monochrome + Monochrome). Außerdem steht ein BAS und Tonsignal getrennt zur Verfügung. Der Ton ist brumm- und rauschfrei! Die Ansteuerung von **U2** ist vollkompatibel zu anderen auf dem Markt befindlichen Auto-Monitor-Switchboxen. Ein Ansteuerbeispiel liegt bei.
Superpreis: nur 39,90 DM!!

Zusätzlich bieten wir **U2** auch als Bausatz an.
Komplett ohne Gehäuse: **DM 24,-**
Komplett mit Gehäuse: **DM 29,-**
Gehäuse gebohrt mit Cinchbuchsen.

Ramkarte **nur DM 49,-**
Leerkarte für Ram-Erweiterung auf 1 MB, 13 Lötpunkte notwendig. Voll gesockelt, Platine mit Industriequalität. Passend für ST 260/520/520+



MARS ST **nur DM 89,-**
Das Strategiespiel für Computerfans. Der Computer ist die Arena, die Kämpfer entspringen Ihrer Phantasie. Voller Weltmeisterschaftsstandard, integrierter Editor, gutes Handbuch. ASM-Hit 12/87. Demoversion gegen DM 10,- Vorausscheck.

Sampler
Mit den Samplern von GALACTIC können Sie die Einführung einer neuen Qualitätsstufe auf dem Heimsektor erleben. Alle unsere Geräte beherrschen in der Wiedergabe Oversampling zur Verringerung des störenden Sirrens (Modulationsrauschen). Zusätzlich kann bei allen Geräten direkt in 4 Bit gesampelt werden, dadurch können doppelt so lange Stücke gesampelt werden wie bisher. Unsere Profigeräte verfügen zusätzlich über Zusatzschaltungen, die fast allen Samplerfehlern beikommmen können. Ein rauhes oder metallisches Klangbild gehört der Vergangenheit an. Außerdem lassen sie Hinterbandkontrolle vor und während der Aufnahme in 1:1-Qualität zu (**WYHIWYG**, **W**hat **Y**ou **H**ear **I**s**W**hat **Y**ou **G**et), wie man es von professionellen Studiobandmaschinen her kennt. Auch über Midifähigkeiten verfügen diese beiden Geräte.

Volkssampler: nur Monitor **DM 99,-**
Volkssampler +: Wiedergabe über Monitor und Wandler **DM 169,-**
Prosampler 8 mono: Profigerät höchster Klanggüte **DM 298,-**
Prosampler 8 stereo: erstmals echtes Stereo!! **DM 398,-**

Autoren gesucht!

Haben Sie ein interessantes Programm geschrieben oder ein gutes Hardwareprojekt entwickelt, wenden Sie sich bitte an uns.

Roboterinterface
Mit diesem Interface für die Roboter SVI-2000 können Sie erstmals zu einem günstigen Preis Robotertechnologie auf dem Atari simulieren. Einfach aus jeder Programmiersprache heraus anzusteuern können Sie Objekte manipulieren und bewegen. Auch für Demos und Vorführungen bestens geeignet. Der Betrieb erfolgt über Batterien, kann aber auch über Standardsteckernetzteile erfolgen. **nur DM 98,-**



DEEP THOUGHT
neue Version
Farbe & Monochrome
Unser bewährtes Schachprogramm liegt in einer neuen Version vor. Die frei programmierbare Eröffnungsbibliothek und der spielstarke Algorithmus haben es beliebt gemacht. Die einzigartigen Funktionen wie Blitzstufe und Partienarchivierung mit Zusatzdaten machen es für den ernsthaften Spieler zu einem wichtigen Arbeitsutensil. **nur DM 69,-**

Zusätzlich im Angebot!
Zusätzlich im Angebot: Kabel, Stecker, Buchsen und vieles mehr. Fordern Sie ausführliche Infos an.

Versandbedingungen:
Inland: Nachnahme DM 7,50 Porto und Verpackung
Vorkasse DM 4,50
Ausland: nur Vorkasse rein Netto + DM 6,50
Ab DM 500,- Warenwert Versandkostenfrei.
Auf alle Produkte ein Jahr Garantie!

GALACTIC · Stachowiak, Dörnenburg und Raeker GbR
Burggrafenstraße 88 · 4300 Essen 1 · ☎ (0201) 273290/710 18 30 · Telefax (0201) 710 19 50
Bankverbindungen: Sparkasse Essen (BLZ 360 501 05) Kto.-Nr. 3712 056 · Postgiroamt Essen (BLZ 360 100 43) Kto.-Nr. 1998 72-435

BILDWERKSTATT ATARI ST

Videotechnik und Computer

Im ersten Teil unserer Serie haben wir versucht, Ihnen einen kleinen Einblick in die Welt der Computergrafik zu geben, Ihnen zu zeigen, was bis heute bereits auf diesem Gebiet erreicht wurde. Daß dies wirklich nur ein ganz kleiner Einblick sein konnte, liegt daran, daß das Anwendungsfeld für Computergrafik ständig größer wird:

Das Modewort "Desktop Publishing" ...

1985 begann die amerikanische Firma APPLE auf dem Macintosh-Computer mit der Vorstellung von *"Desktop Publishing"*. DTP ist, kurz gesagt, ein System zur computerunterstützten Gestaltung von Textseiten. Dazu gehört sowohl die Einbindung von Grafiken in den Text, als auch Spaltensatz und die Auswahl verschiedener Schriftarten und -größen. Doch während DTP- (oder SAP-) Programme auf dem ATARI ST zur Zeit einen Höhepunkt erleben, beginnt sich der Trend auf dem Macintosh-Computer bereits wieder zu wenden: Programme für Präsentationsgrafik sind dort jetzt groß im Kommen.

Ein Beispiel: Nach einer amerikanischen Marktstudie besteht bei nur 17 % der großen und mittleren Unternehmen ein Bedarf an DTP-Produkten, Präsentationsgrafik zur Umsetzung von trockenen Zahlen und Fakten zur Darstellung eigener Ideen als Computergrafiken wird jedoch von rund 72 % der Unternehmen verlangt.

(Quelle: Computer Persönlich vom 14.9.88).



"Desktop Video" - ein neues Modewort?

Die Anwendungen von Computergrafik sind so vielseitig, daß man sie einfach nicht mehr aufzählen kann.

Von der Darstellung mathematischer Gesetze (Beispiel: Fraktalgrafiken) über Wahlanalysen in Form von Tortengrafiken oder Balkendiagrammen, wie wir sie aus dem Fernsehen kennen, aufsehenerregenden Computeranimationen aus der Werbebranche, aufwendigen Umweltsimulationen bis hin zum Einsatz des Computers in der Kunst reicht das Spektrum der Anwendungen.

Völlig neue Dimensionen eröffnen sich, wenn wir die bisherige Videotechnik und die neue Computertechnik der digitalen Bildverarbeitung miteinander verbinden: Die Zusammenarbeit beider Systeme wird seit einiger Zeit *"Desktop Video"* genannt. Und wir möchten Ihnen jetzt zeigen, was man mit dieser Kombination bereits alles anfangen kann:

Anm.: Unter digitaler Bildverarbeitung wird oft auch nur verstanden, Bildinformationen mit Hilfe von rechnerunterstützten Filterungen zu verarbeiten, beispielweise, um den Kontrast eines Computerbildes zu verändern. Auch diese Techniken werden wir im Rahmen dieser Serie vorstellen.

Jede Menge Bilder

Für die Erstellung von Computerfilmen benötigen Sie jede Menge Bildmaterial:

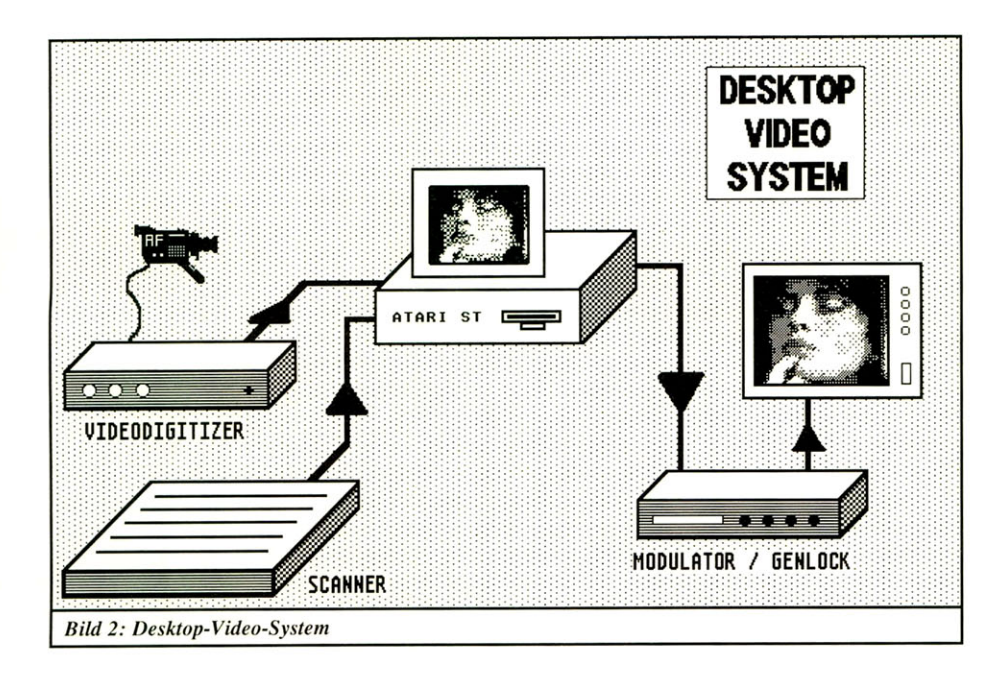


Bild 2: Desktop-Video-System

So wie ein Kinofilm sich aus 18 bis 25 einzelnen Bildern je Sekunde Film zusammensetzt, benötigt auch Ihr Computerfilm eine ganze Reihe von Einzelbildern. Für eine ruckfreie Bewegung sollte es ein Minimum von 10 Bildern für jede Sekunde Animation, die Sie zeigen wollen, sein. Dies entspricht in etwa auch der Anzahl von Bildern je Sekunde, wie sie bei der Erstellung von Zeichentrickfilmen eingesetzt wird. Für eine völlig flüssige Bewegung müssen Sie sogar bis zu 25 Bilder je Sekunde zur Verfügung stellen, was dann der Bildfolgefrequenz von Video entspricht.

Doch wie kommen Sie an das Bildmaterial? In den allerwenigsten Fällen werden Sie so viele Einzelbilder Schritt für Schritt mit einem einfachen Grafikprogramm von Hand zeichnen. Deswegen gibt es inzwischen Programme am Markt, mit denen es möglich ist, eine ganze Reihe von Bildern vom Computer selbständig errechnen zu lassen. Möchten Sie zum Beispiel dreidimensionale Objekte im Raum drehen oder bewegen, so bieten Ihnen Programme wie STAD oder CAD-3D die Möglichkeit, die einzelnen Phasen einer solchen Bewegung in einzelne Bilder zu zerlegen und abzuspeichern.

Videodigitalisierung ...

Einen ganz anderen Weg geht man, indem man Bildvorlagen oder Videofilme mit Hilfe eines "Videodigitizers" in den Speicher des Rechners einliest. Dabei wird ein analoges Bildsignal, das von einer Videokamera oder von einem Videorecorder kommt, mit einem Hardwarezusatz in digitale Informationen zerlegt und dann in Form binärer Daten im Rechnerspeicher abgelegt. Wie das im einzelnen funktioniert, werden wir später noch zeigen.

... und Scannen liefern die Bilderflut

Ebenfalls möglich ist die direkte Abtastung einer Bildvorlage über ein lichtempfindliches Element. Diesen Vorgang nennt man *"Scannen"*. Im Gegensatz zur Digitalisierung fällt die gesamte Videotechnik komplett weg, allerdings können die Bilder meist nur in Schwarzweiß in den Rechner eingelesen werden, Scanner liefern teilweise auch keine Grauwerte der Bildinformation an den Rechner, sondern nur reine Schwarzweiß-Raster. Die Auflösung feiner Strukturen ist beim Scannen jedoch um einiges besser als bei der Videodigitalisierung.

Für die Abtastung von Schwarzweiß-Vorlagen wie Skizzen, Emblemen und Texten ist die Scannertechnik weit besser geeignet als die Videodigitalisierung. Für farbige Vorlagen, Videofilme oder in Helligkeit und Farbe fein abgestufte Bildvorlagen ist die Digitalisierung unentbehrlich.

Wir möchten auch darauf hinweisen, daß die Grenzen beider Techniken fließend sind: Einerseits gibt es Videodigitizer in professioneller Ausführung mit sehr feiner Auflösung, andererseits erlauben auch viele Scanner eine Erkennung von Grauwerten, sogar Farbscanner sind erhältilch. Doch beidesmal hat die hohe Qualität auch ihren Preis: Die Geräte sind deutlich teurer und eigentlich nur für professionelle Anwender gedacht.

Die Bildwerkstatt

Nachdem die einzelnen Bildvorlagen in Rohform in den Rechner eingelesen wurden, kann die Bearbeitung im Rechner erfolgen. Hier kommt der interessanteste Teil der gesamten Arbeit. Der Rechner ist elektronische Schere und Zeichenstift zugleich, kann wie eine nachträgliche Zoomvorrichtung oder ein Farbfilter eingesetzt werden.

Überlagerung von Bildern, Collagen, Farbanpassung, Spiegelung und Beschriftung - den Möglichkeiten der elektronischen Verarbeitung sind eigentlich kaum Grenzen gesetzt, und wir werden uns in dieser Serie ausführlich mit diesen Techniken befassen.

In der nächsten Phase zu unserer Animation steht uns der Computer dann als Regieassistent zur Verfügung: Wie der Schritt von der Fotografie zum Film, so wird der Computer vom Diabetrachter zum Projektor. Und dabei ist er zugleich mehr als nur ein elektronisches Daumenkino:

Die Einzelbilder aus dem Rechnerspeicher muß man nicht einfach nur Bild für Bild "abspielen", in *IMAGIC* bepielsweise sind viele Effekte für eine Überblendung von Bild zu Bild bereits vorprogrammiert, der Computer errechnet einzelne Phasen oder Bewegungen in Echtzeit! Damit können dann mehrere Sekunden Film erstellt werden, ohne daß Dutzende von Einzelbildern benötigt werden. Diese Technik hat auch ihre Grenzen: Vergrößerung und Verkleinerung oder Drehungen von ganzen Bildern lassen sich in Echtzeit auf einem ATARI ST nicht mehr berechnen. Hier werden die Phasen nach wie vor als Einzelbilder gespeichert und dann in schneller Folge gezeigt.

... und zurück auf Video

...wenn es beliebt. Denn es ist durchaus möglich, komplette Filme auch im Rechnerspeicher vorzuführen, wenn die Bilder nur genügend dicht komprimiert und auch schnell genug wieder ausgepackt werden können.

Dazu eine kleine Anmerkung: Wer sich speziell für die Technik der Bildkomprimierung interessiert, den möchten wir gern auf die Serie "BILDUNG" *von Stefan Höhn in der* ST-Ecke *dieser Zeitschrift hinweisen. (ST-COMPUTER, ab September 1988).*

Doch wir nehmen einmal an, Sie möchten

einen Titelvorspann für ein selbstgedrehtes Urlaubsvideo erstellen: Dann müssen Sie das Ergebnis irgendwann natürlich zurück auf Video überspielen, am besten vielleicht sogar in einen laufenden Videofilm einblenden.

Zur einfachen Überspielung benötigen Sie einen *Modulator*, der das *RGB-Signal* des ATARI ST in ein Videosignal verwandelt, das Ihr Videorecorder versteht. Der ATARI 512 STM hat einen solchen Modulator bereits fest eingebaut, allerdings mit einem HF-Signalausgang für den direkten Anschluß an die Antennenbuchse eines Fernsehgerätes. Dieses HF-Signal geht aber eine Stufe zu weit, der Videorecorder muß es wieder zurückwandeln in *"FBAS"*, denn nur dieses Signal kann er auf Magnetband aufzeichnen. (Einige wenige Videorecorder zeichnen auch direkt ein RGB-Signal auf, hier erübrigt sich ein Modulator).

Eine weit bessere Qualität jedoch erhalten Sie mit einem Modulator, der das Signal *FBAS* erzeugt und auch an einer Buchse nach außen führt. Dieses Signal verstehen fast alle Videorecorder direkt, die Qualität des erzeugten Videobildes ist deutlich besser. Aus eigener Erfahrung möchten wir Ihnen jedoch raten, sich die Bildqualität eines Modulators vor dem Kauf anzusehen. Es gibt hier Qualitätsunterschiede, die Sie am Preis des Gerätes nicht unbedingt erkennen können.

Der große Clou bei der Überspielung von Computerbildern auf Video sind natürlich sogenannte *"GENLOCK-Interfaces"*. Diese Geräte erlauben die Mischung von bewegten Videobildern mit Grafiken aus dem Computer. Dabei gibt es preiswerte Geräte, die nur die Farbe SCHWARZ aus der ATARI-Grafik durch das Videobild ersetzen. Die professionellen Versionen erlauben dabei sogar das Ausblenden ganzer Farbbereiche und die Einmischung von 2 unterschiedlichen Videoquellen gleichzeitig.

Anmerkung: Die Aufzeichnung von Computergrafiken auf Video ist grundsätzlich nur in den beiden Farbauflösungen des ATARI ST möglich. In der hohen Auflösung Schwarzweiß arbeitet der Rechner mit einer deutlich höheren Bildfrequenz, die Signale lassen sich nicht auf Video aufzeichnen. Auch ein Abfilmen des Monitors ist nicht empfehlenswert, da die Differenz der Bildfrequenzen Störstreifen im Videobild erzeugt, die man beim einfachen Betrachten des Monitorbildes nicht sieht. Natürlich kann man trotzdem Computerfilme in Schwarzweiß erzeugen, man kann sie eben nur im Computer ansehen ...

Soviel also ungefähr zur erforderlichen Hardware. Wir gehen später noch einmal etwas genauer auf die benötigten Geräte ein. Zusammen mit einem kleinen Farbfernseher als Kontrollmonitor benötigen Sie dann sicherlich auch noch einen größeren Tisch oder eine Arbeitsecke im Hobbykeller, um alle Aufbauten unterzubringen.

Doch jetzt geht es ein wenig mehr ins Detail:
Einige Fachbegriffe wie *"RGB-Signal"*, *"Abtastung"*, *"Bildauflösung"* usw. möchten wir Ihnen gerne näher erläutern.

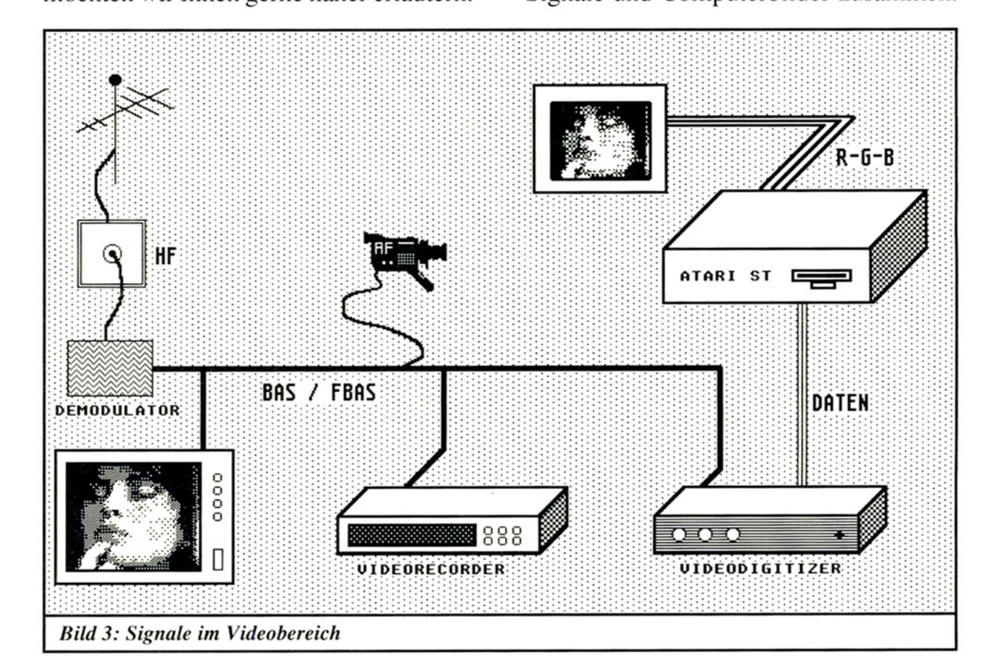


Bild 3: Signale im Videobereich

Horch, was kommt von draußen rein

Vom Antennensignal zum Fernsehbild

Fangen wir ganz vorne an: Ein Antennensignal, so wie Sie es an der Antennenbuchse Ihrer Hausantenne oder an der HF-Buchse eines 520 STM entnehmen können, nennen wir ein *"Hochfrequenzsignal"*, kurz *"HF"*. Das elektrische Signal hat hier noch eine Frequenz von einigen hundert Millionen Schwingungen pro Sekunde. In den Schwingungen sind aber bereits die eigentlichen Bild- und Toninformationen des Fernsehbildes enthalten, verschlüsselt in kleinen Intensitäts- und Frequenzschwankungen des HF-Signals.

Ihr Fernsehgerät oder Videorecorder nimmt in seinem Empfangsteil jetzt die erste Phase der Entschlüsselung vor: Ein sogenannter DEMODULATOR filtert die extrem hohen Schwingungen, die eigentlich nur dem Transport der Information gedient haben, aus dem HF-Signal aus. Als Ergebnis erhalten wir das reine Videosignal, genannt *"Bild-Austast-Synchronsignal"*, kurz *"BAS"*, das alle Bild- und Toninformationen sowie die Synchronisationsbefehle für den Bildaufbau noch gemeinsam enthält. Ist zusätzlich auch eine Farbinformation enthalten, nennt man das Signal *"FBAS"*.

Dies ist die gemeinsame Basis zur Übermittlung von Bildinformationen in den Computer und aus dem Computer heraus: Ein Videodigitizer tastet *BAS* oder *FBAS* ab, eine Farb-Videokamera liefert *FBAS*, ein *GENLOCK-Interface* mischt *FBAS*-Signale und Computerbilder zusammen. Und mit einem Videorecorder kann man *FBAS* aufzeichnen oder abspielen.

In einer nächsten Stufe werden die im *BAS* oder *FBAS* enthaltenen Informationen voneinander getrennt: Ton, Synchronisation, Bildsignal. Bei *FBAS* entstehen sogar gleich dreimal Bildsignale: Getrennt für *Rot*, *Grün* und *Blau*, im Ganzen werden sie auch *RGB*-Signal genannt. Diese Bildsignale gelangen im Fernseher dann über einen Verstärker direkt zur Ansteuerung der Bildröhre: Das Fernsehbild ist da!

Dies übrigens ist auch die Ebene, auf der unser ATARI ST seine Farbbilder verwaltet: Alle 512 möglichen Farben, die ein ATARI ST ohne spezielle Softwarekniffe erzeugen kann, werden aus Rot, Grün und Blau zusammengemischt. An der Monitorbuchse Ihres ATARI können Sie das RGB-Signal abgreifen, vorausgesetzt, Sie arbeiten im Farbmodus.



Jetzt beim Zeitschriften-händler

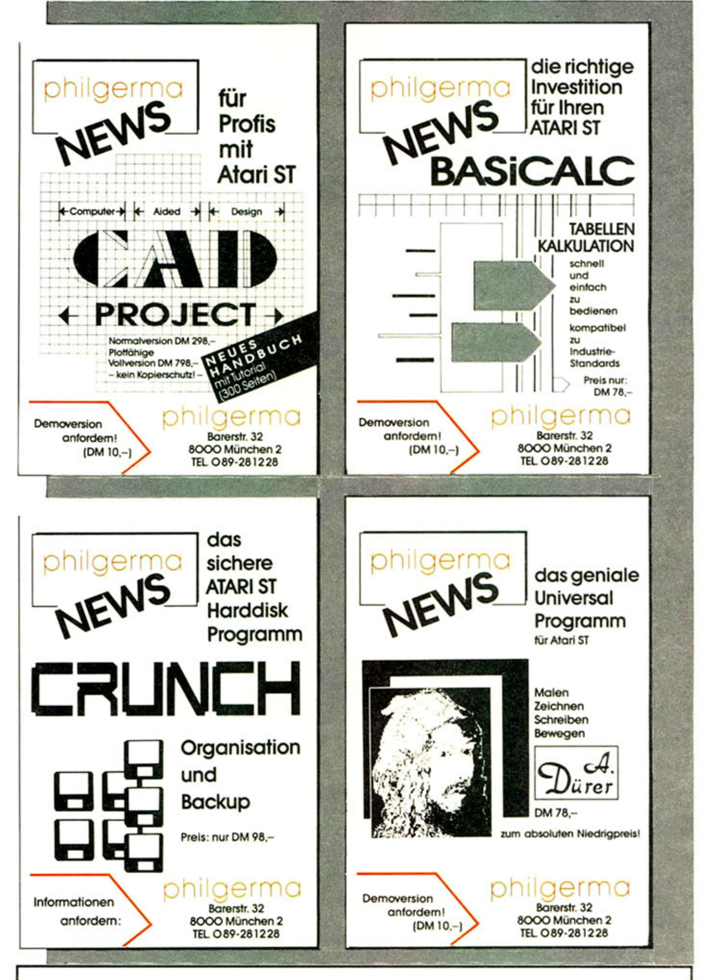
ST DIGITAL bringt jeden Monat aktuelle Softwaretest, Tips, Tricks und vieles mehr, was ATARI-Fans wissen müssen.

Lesen Sie im Dezemberheft:

■ **STOS** Demo auf Diskette ■ **MASTERPAINT** – Grafiktool für Profis ■ **NEWS** – ATARI-Messe PCW-SHOW London ■ **TEST** – HAWK CP 14, Scanner für Kenner ■ **CAD 3D** – Animation Live



Jetzt beim Zeitschriftenhändler





Unser Service endet nicht an der Ladentür. Auch bei Versand-bestellung garantieren wir Ihnen unsere volle Unterstützung.

Aus unserem umfangreichen Software-Angebot:

SPRACHEN

Produkt	DM
Lattice C dt. Handb.	298
MCC Assembler neu V12	248
MCC Pascal 2 dt. Handbuch	248
Pro Pascal	248
Pro Fortran 77	378
SPC Modula 2	348
AC Fortran 77	448
Aztek C professional	398
Aztek C developer	598
Aztek C S. L. Debugger	148
Mark-Williams C V3.0	298
Turbo C	198
Salix Prolog 2	198
Omikron Basic Compiler	178
GFA Basic Interp. V 3.0	198
Cobol-Paket + PC-Ditto	398
OS-9 m. div. Compilern	1598

TEXT

Produkt	DM
WordPerfekt Textpr. eng.	798
Signum II Textgestalt.	428
StarWriter ST Textpr. dt.	198
Tempus 2.0 Editor	109

GRAFIK

Produkt	DM
Dürer s/w Malprogramm	78
Spektrum 512 farb. Malpr.	148
CADproject Normalversion	298
CAD-3D Cyberstudio	178
Calamus DTP	398

Produkt	DM
Aladin neu V2.1 + Rom's	598
BASiCALC Tabellenkalk.	78
K-Graph 3 Grafik u. Stat.	198
Querdruck f. Tabellenkalk.	58
Anti Viren Kit	78
ST-MATH Algebra + Analysis	98
K-Comm 2 Terminalprogramm	148
Logistix integr. Paket	398
dBMAN V 4.0 Datenbank	598
ST Base III Datenbank	698
TIM II Buchführung	598
fibuMAN f Buchhaltung	798
BTX-Manager V3.0	428
PC-DITTO MS-DOS-Emulator	198

SPIELE

Produkt	DM
Psion Chess s/w u. Farbe	69
Carrier Command F.	69
Bolo s/w	69
Bubble Bobble F.	59
Gauntlet II F.	69
Flight II Flugsim. s/w u. F.	99
European Scenery Disk	59
Jet Flugsim. F16 und F18	99

HARDWARE

Produkt	DM
Harddiskcontroller PC-Hardd.	348
Einzellaufwerk 3.5"	298
Einzellaufwerk 5.25", 40/80	398
10 Disketten 3.5", 2DD Fuji	36
10 Disket. 3.5", 2DD no name	26

Preis- oder Händlerlisten anfordern!
Telefonische Bestellannahme und Hotline-Service: 089/28 12 28
Bei Bestellungen unter DM 200,– beträgt der Versandkostenanteil DM 4,80. Nachnahme DM 3,20. Ins Ausland liefern wir nur gegen Vorkasse (Überweisung oder Euroscheck).

philgerma
Barerstr. 32 · 8000 München 2
☎ 089-281228

Neu: 4600 Dortmund 50
Baroperstr. 337, ☎ 0231-759292

Tastversuche

Betrachten wir ein reales "Bild" aus unserer Umwelt, so hat es eine Strukturfeinheit und eine Farbenvielfalt, die nur durch die Anzahl der Rezeptoren unserer Netzhaut begrenzt ist. Computerbilder sind jedoch meist viel gröber. Die Anzahl der Informationen pro Bild muß für den Computer drastisch reduziert werden, einerseits, um den Speicherplatz pro Bild in Grenzen zu halten, andererseits auch, weil die zur Verfügung stehnde Hardware im Computer zur Darstellung der Bildinformation keine feineren Abstufungen erlaubt.

Ein Standard-Computerbild im ATARI ST hat unterschiedliche Auflösungen: Von 256000 Punkten in Monochrom bei der hohen Auflösung bis zu 64000 Punkten in 16 Farben bei niedriger Auflösung.

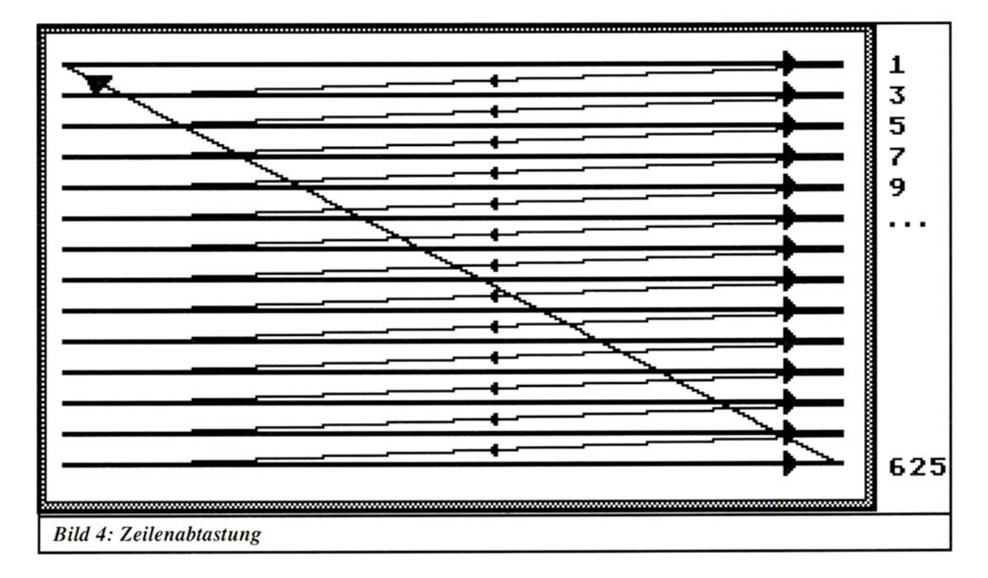


Bild 4: Zeilenabtastung

Wie jedoch wird ein "reales" Bild in ein solches Computerbild gewandelt? In einem ersten Schritt zerlegt die Videokamera ein beliebiges Bild aus unserer Umwelt in genau 625 schmale horizontale Streifen. Diese Streifen werden dann einzeln nacheinander von links nach rechts abgetastet. So entsteht aus einem zweidimensionalen Abbild der Wirklichkeit eine eindimensionale Kette von Bildinformationen. Da das Verfahren folglich eine Bildzeile nach der anderen abtastet, nennt man dieses Abtastschema "Zeilenabtastung".

Genau genommen werden bei der Zeilenabtastung eines Fernsehbildes erst in einem Durchlauf die geraden Bildzeilen und dann in einem weiteren Durchlauf die ungeraden Bildzeilen abgetastet. Dadurch wird das Bild in zwei "Halbbilder" *zerlegt. Der Sinn der Methode: Bei der entgegengesetzten Rekonstruktion des Ursprungsbilds später in der Bildröhre wird das Bildflackern deutlich reduziert. Diese Methode verdoppelt nämlich die Bildfrequenz, ohne dabei die Informationsmenge zu verdoppeln.*

Die "abgespulte" Kette der Bildinformationen wird jetzt noch technisch aufbereitet: Handelt es sich bei der Videokamera um eine Farbkamera, werden Farb- und Helligkeitsinformation je Bildpunkt von der Kamera getrennt. Nach dem *PAL*-Verfahren wird das Helligkeitssignal mit einer aufmodulierten Farbinformation versehen, die dem Farbwert des einzelnen Bildpunktes entspricht. Ergänzt wird die reine Bildinformation jetzt noch um die Taktsignale "nächste Zeile" (horizontale Synchronisation) und "nächstes Bild" (vertikale Synchronisation). Das Ergebnis stellt dann das oben bereits erwähnte *BAS*- oder *FBAS*-Signal dar.

Anmerkung: Vereinfacht dargestellt bewirkt die Farbinformation im Bild also eine Art "Phasenverschiebung" *des Helligkeitssignals für jeden Bildpunkt. Dabei tritt das Signal gegenüber einem festen Vergleichstakt geringfügig früher oder später auf. Im Zusammenhang mit der Videodigitalisierung ist diese* "Phasenverschiebung" *von Bedeutung, weil sie zu Störungen führt.*

Videodigitalisierung

Noch liegen uns die Bildinformationen als analoge Signale vor, die Helligkeits- und Farbwerte einzelner Bildpunkte haben noch beinahe unendlich feine Abstufungen. Um die Bildinformation jedoch in einem Computer verarbeiten zu können, müssen die Punkte "digitalisiert" werden. Das geschieht in zwei Schritten: Zuerst wird das Zeilensignal mit einem festen Takt abgetastet. Dabei werden zu genau definierten Zeitpunkten *"Proben"* der Bildinformation entnommen. Diesen Vorgang nennt man *"Abtastung"*, die Anzahl der Abtastpunkte je Bildzeile *"Rasterung"*. Die Bildauflösung eines Videodigitizers ergibt sich aus der Anzahl von Punkten je Zeile und Zeilen je Bild, die das Gerät dabei erfassen kann.

Der zweite Schritt ist die *"Quantisierung"*. Das bedeutet einfach, daß die gesamte Skala möglicher Helligkeits- und Farbwerte je abgetastetem Bildpunkt in feste Stufen unterteilt wird. Für jeden Bildpunkt wird dann aus dieser *"Treppe"* der feste Wert ausgewählt, der dem realen Wert am nächsten kommt. Dazu werden sogenannte *Analog-Digitalwandler* (AD-WANDLER) eingesetzt. Als Ergebnis erhält man eine Folge von einzelnen Werten, die als digitale Daten direkt in den Rechner eingelesen werden können.

Die Qualität der Videodigitalisierung wird also hauptsächlich durch den eingesetzten Typ des AD-WANDLERS bestimmt. Dabei gibt es große Unterschiede: Es gibt Videodigitizer, die besitzen nur 4 Stufen für den Grauwert eines Bildpunktes bei der Abtastung, zeichnen sich aber dafür durch eine besondere Schnelligkeit aus, weil die 4 Stufen in einem Schritt erkannt werden.

In den meisten Fällen wird für den ATARI ST ein 4-Bit AD-WANDLER eingesetzt, der 16 verschiedene Helligkeitswerte je Bildpunkt liefert. Solche AD-WANDLER sind jedoch langsamer als eine einfache Abtastung, deswegen lassen sich Bilder oft nicht mehr in Echtzeit digitalisieren, die Qualität der digitalisierten Bilder ist hier jedoch wiederum deutlich besser. Für eine Echtfarbendigitalisierung ist der Einsatz von Digitizern mit mindestens 16 Graustufen unserer Meinung nach unentbehrlich.

Zusammengefaßt kann man die für den ATARI ST erhältlichen Videodigitizer in zwei Gruppen einteilen:

- *Echtzeitdigitizer*, die 25 Bilder je Sekunde erfassen können. Die hohe Geschwindigkeit wird jedoch oft mit einer reduzierten Anzahl von Grauwerten und einer geringeren Auflösung erkauft. Es gibt allerdings Geräte, die setzen einen besonders schnellen AD-WANDLER, einen "FLASH-AD-WANDLER" ein. Diese Geräte liefern genügend Grauwerte, sind allerdings, wie so oft, besonders teuer.

- *Digitizer mit hoher Auflösung*, die meist

nicht in Echtzeit arbeiten, dafür aber weit bessere Bilder liefern. Dabei ist es durchaus von Vorteil, wenn ein Digitizer eine höhere Auflösung hat, als Ihr ATARI ST eigentlich darstellen kann. Die höhere Auflösung liefert zusätzliche Informationen, die von einer guten Software genutzt werden können, um feinere Farb- und Helligkeitsverläufe im Computerbild zu erzeugen.

Echtzeitdigitizer sind immer dann notwendig, wenn Sie großen Wert auf Geschwindigkeit bei der Digitalisierung legen: wenn Sie beispielsweise "live" direkt von einer Videokamera eine ganze Sequenz von Einzelbildern digitalisieren und abspeichern wollen. Sehr wichtig ist die Echtzeitdigitalisierung natürlich bei der Mustererkennung, z.B. in automatischen Sortieranlagen: Dort "sieht" ein Computer mit Hilfe einer Videokamera und einer Echtzeitdigitalisierung, vergleicht die eingelesenen Bilder mit gespeicherten Mustervorlagen und kann daraufhin beispielsweise Werkstücke sortieren etc.

Für die Erstellung von Computerfilmen ist es unserer Meinung nach vorteilhafter, Bildvorlagen auf einen Videorecorder aufzuzeichnen und später im Standbildmodus des Recorders Bild für Bild einzeln zu digitalisieren. In diesem Fall ist ein Digitizer mit hoher Auflösung einem schnellen Digitizer vorzuziehen. Hat Ihr Recorder ein gutes Standbild, ermöglicht Ihnen das die Einzelbilddigitalisierung von fertigen Videofilmen. Die Bildwiedergabe beim Standbild sollte dabei frei von irgendwelchen Störstreifen sein. Ein leichtes Flimmern oder Zittern des Bildes ist dabei wiederum nach unserer Erfahrung nicht von Nachteil: Dieses Zittern entsteht erst im Fernsehgerät durch leichte Schwankungen der Bildsynchronisation. Ein Videodigitizer liefert trotzdem einwandfreie Bilder.

Farbenzauber

Ein wichtiges Kapitel bei der Erstellung von Computerfilmen ist die Echtfarbendigitalisierung. Es ist unseres Wissens noch kein Digitizer für den ATARI ST verfügbar, der die PAL-Farbinformation aus dem FBAS-Signal an seinem Eingang auswertet, und dem Computer daraufhin direkt Informationen über Helligkeit und Farbe des Videobildes zur Verfügung stellt. Im Gegenteil: Für viele Digitizer ist die Farbinformation im Videosignal sogar störend: Die im Bildsignal enthaltene Farbinformation macht sich als unschönes *Moiré-Muster* im digitalisierten Bild bemerkbar.

Abhilfe schafft hier nur die Ausfilterung der Farbinformation aus dem *FBAS*-Signal. Wird das nicht vom Digitizer selbst erledigt, haben Sie mehrere Möglichkeiten:

1. Sie verwenden eine Schwarzweiß-Videokamera. Sie liefert grundsätzlich nur *BAS*-Signale, die den Digitizer nicht stören. Mit Hilfe der Farbscheiben-Methode können Sie auch hier Echtfarbbilder digitalisieren (s.u.).

2. Sie verwenden einen Videorecorder, der die Möglichkeit bietet, Filme auch in Schwarzweiß abzuspielen. Leider bieten nur sehr wenige Recorder diese Möglichkeit.

3. Sie schalten einen *Farbträgersperrfilter* vor Ihren Videodigitizer. So ein Filter wandelt ein *FBAS*- in ein *BAS*-Signal um.

4. Sie setzen einen sogenannten *"RGB-Splitter"* ein, der das Videosignal auch gleich für die Echtfarbendigitalisierung aufbereitet. Nur so ist es möglich, Videofilme in Echtfarben zu digitalisieren.

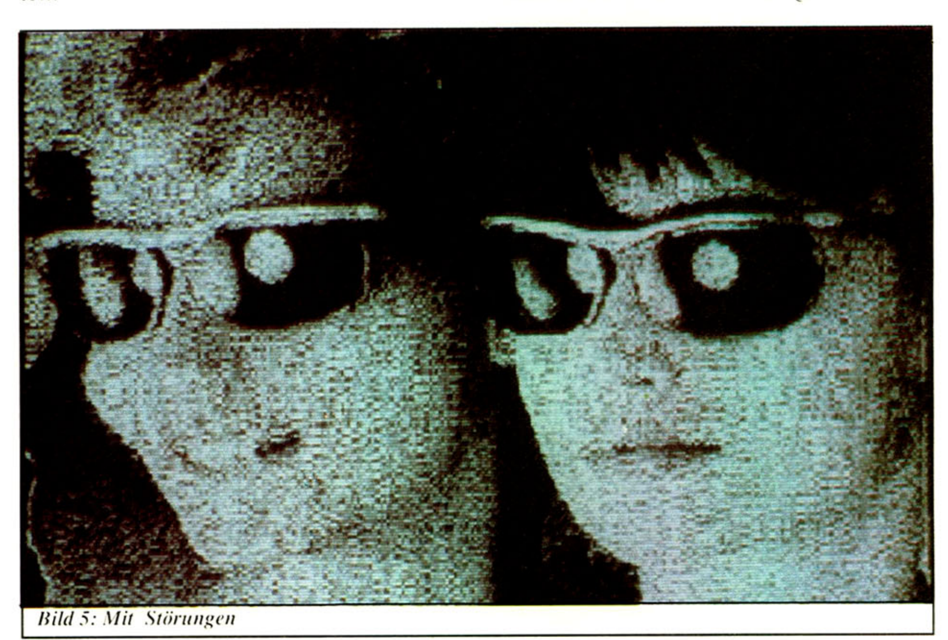

Bild 5: Mit Störungen


Bild 6: Ohne Störungen

Als Voraussetzung zur Echtfarbendigitalisierung muß das farbige Videobild erst einmal in drei gefilterte Bilder zerlegt werden. Als einfache Methode, und wenn Sie eine Videokamera einsetzen, gibt es die Technik, die Farbanteile von *ROT*, *GRÜN* und *BLAU* mittels Farbscheiben auszufiltern. Dazu werden drei farbige Filterscheiben abwechselnd direkt vor die Linse der Videokamera gehalten und dann die Digitalisierung aller drei Bilder

einzeln durchgeführt. Auch hier wiederum ist es empfehlenswert, das Farbmoiré auszufiltern, wenn Sie eine Farbkamera einsetzen.

Ein *"RGB-Splitter"* spaltet ein *FBAS*-Signal auf in jeweils ein *ROT-*, *GRÜN-* und *BLAU*-Signal, das anschließend digitalisiert wird. Diese Technik erlaubt, wie erwähnt, sogar die farbige Digitalisierung von Viodeofilmen.

Mit etwas Rechenaufwand entsteht jetzt das Farbbild im Computer: Ein Programm ermittelt aus den digitalisierten Daten für jeden Bildpunkt die benötigte Mischung aus *ROT-*, *GRÜN-* und *BLAU*-Anteilen. Gleichzeitig wird eine optimale *Farbpalette* ermittelt, in der die am meisten im Bild vorkommenden Farben enthalten sind. Dieser Rechenvorgang entspricht dabei meist einer Reduzierung der zur Verfügung stehenden Informationen zu einem einzigen farbigen Computerbild.



In der Praxis

Doch jetzt genug von Theorie und Fachsimpelei, kommen wir zur Praxis der Videodigitalisierung:

PUNKT 1
WAS BENÖTIGEN SIE?

Für die Digitalsierung gedruckter Vorlagen wie Fotos oder Prospekte sollten Sie ein *Reprostativ* besitzen. Damit können Sie den Abstand von der Vorlage zur Videokamera exakt einstellen, auch läßt sich die Vorlage genau positionieren. Zur Digitalisisierung von Vorlagen, die größer als das Format DIN A4 sind, sollte das Stativ eine Höhe von mindestens einem Meter haben. Ersatzweise können Sie auch ein *Dreibeinstativ* einsetzen, das Sie zu Ihrer Videoausrüstung eventuell bereits besitzen.

Weiterhin sollte eine Videokamera vorhanden sein, am besten mit einer *MAKRO*-Einrichtung, damit Sie einer Vorlage auch einmal ganz dicht "zu Leibe" rücken können. Achten Sie darauf, daß die Kamera eine Bohrung für den Adapter hat, mit dem Sie die Videokamera am Stativ befestigen können.

Eine professionelle Repro-Beleuchtung, wie auf unserem Foto, ist für die Videodigitalisierung nicht unbedingt erforderlich. Eine oder besser zwei Lichtquellen, die aus einer Entfernung von ca. 1-2 Metern auf die Vorlage strahlen, reichen völlig aus. Wichtig ist dabei, daß die Lichtquellen sich nicht in der Vorlage spiegeln. Kontrollieren Sie das im Sucher der Kamera. Eine diffuse, das heißt indirekte, Beleuchtung der Vorlage ergibt bei der Digitalsierung deutlich bessere Ergebnisse. Vermeiden Sie auch eine zu große Helligkeit.

Ein Hinweis: Künstliches Licht kann die Farben bei der Echtfarbdigitalisierung verfälschen. Das gilt insbesondere für Neonlampen. Vergleichen Sie die Farben Ihrer Vorlage mit dem Bild auf Ihrem Kontrollmonitor.

Ein Videorecorder dient Ihnen zur Aufzeichnung oder zum Abspielen der Vorlagen. Falls die Möglichkeit besteht, das Gerät auf Schwarzweißbetrieb umzustellen, sollten Sie das tun (s. Anmerkung *"PAL-Störungen"* oben). In den meisten Fällen werden Sie das Videosignal von der Kamera erst einmal in den Videorecorder einspeisen.

Als Kontrollmonitor können Sie jedes beliebige Farbfernsehgerät einsetzen. Vorteilhaft ist natürlich ein tragbares Gerät, denn sonst müssen Sie schwer schleppen, wenn die Schwarzwaldklinik beginnt ...

Schließen Sie das Fernsehgerät über den *HF*-Ausgang Ihres Recorders an, denn den Ausgang *VIDEO-OUT* benötigen Sie noch für den Digitizer.

Der Videodigitizer wird entsprechend der Betriebsanleitung mit dem Computer und dem Videorecorder / der Videokamera verbunden. Falls Sie jetzt Steckerprobleme bekommen: In vielen Fernsehfachgeschäften erhalten Sie ein Kombi-Pack von Videoadaptern für alle gängigen Kombinationen.

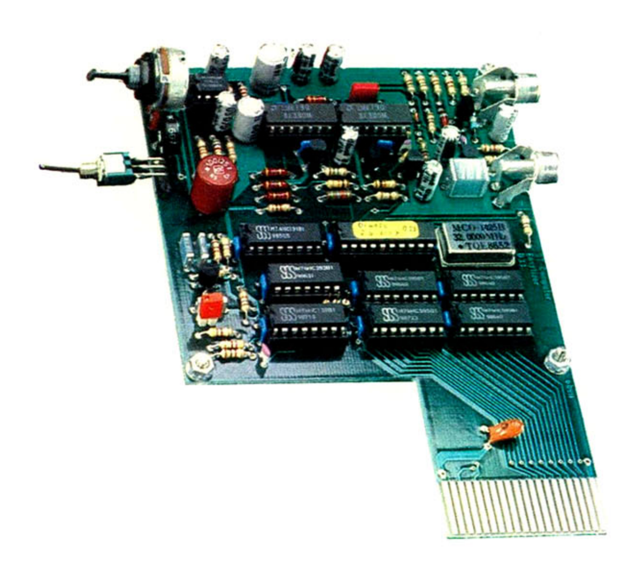
PUNKT 2
WIE WIRD VORBEREITET?

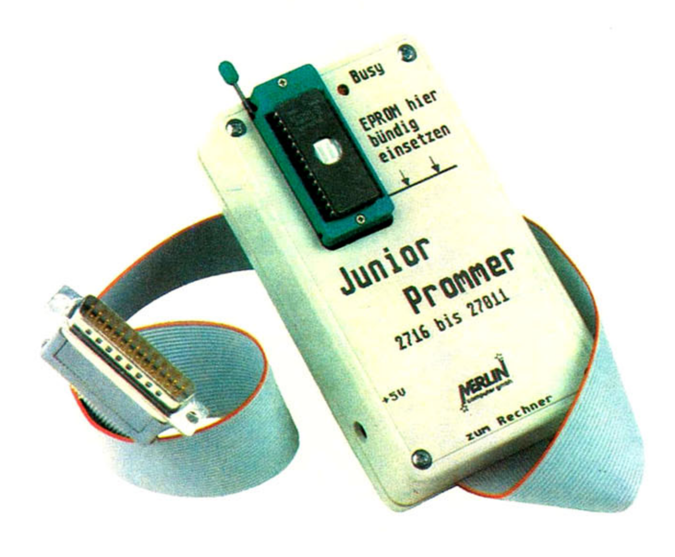
Positionieren Sie Ihre Beleuchtung. Nehmen Sie dazu eine schwarze Vorlage und beobachten Sie das Bild auf dem Kontrollmonitor. Solange sich noch helle Flecken im Bild befinden, haben Sie Reflexionen auf Ihrer Vorlage. Achten Sie allgemein auf eine möglichst gleichverteilte Beleuchtung.

Führen Sie den *Weißabgleich* an Ihrer Videokamera durch. Legen Sie dazu jetzt eine weiße Vorlage unter die Linse der Kamera, die in Ihrem Weißton den Weißtönen Ihrer eigentlichen Bildvorlage entspricht. Bei Prospekten finden Sie den richtigen Weißton oft auf der Rückseite. Haben Sie die Möglichkeit, einen Weißabgleich manuell einzustellen, können Sie auch direkt die Vorlage verwenden, die Sie später digitalisieren wollen. Vergleichen Sie Monitorbild und Vorlage. Stellen Sie lieber etwas kräftigere als zu blasse Farben ein. Achten Sie auf die richtige Stellung des Schalters *"Kunstlicht/Tageslicht"* (wenn vorhanden).

Stellen Sie jetzt die Bildschärfe an Ihrer

Aus unserer Hardwareküche





Easytizer - der Videodigitizer ohne Geheimnisse

Mit dem Easytizer können Sie beliebige Videosignale von einer Schwarzweiß- oder Farb-Kamera, Videorecorder oder direkt vom Fernsehgerät (mit Composite Video Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen.
Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden 12,5 Bilder in der Sekunde wiedergegeben. Im hochaufgelösten Modus werden 640x400 Bildpunkte in Schwarzweiß wiedergegeben.

Besondere Merkmale des Easytizers:

- Software vollständig in Assembler
- Abspeichern der Bilder im DEGAS-Format
- Von STAD und Sympatic Paint ansteuerbar
- Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format vergrößert und verkleinert werden
- Spiegeln eines Bildes in horizontaler und vertikaler Ebene
- Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder)
- Eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible
- Wahlweise automatische oder manuelle Helligkeitseinstellung
- Schnappschuß

Lieferumfang:

1) Fertiggerät
komplett aufgebaut und geprüft, inclusive Diskette mit der Easytizer Software und Bedienungsanleitung
2) Teilsatz
Für Bastler liefern wir einen Teilsatz bestehend aus:
-Doppelseitiger, elektronisch geprüfter Platine mit Lötstoplack und Bestückungskungsaufdruck sowie vergoldeten Anschlußkontakten
- fertig programmiertes GAL 16V8
- Quarzoszillatormodul 32 MHz
- Diskette und Bedienungsanleitung.

Klein, kompakt und leistungsstark- der Junior Prommer

Der Junior Prommer programmiert alle gängigen EPROM-Typen, angefangen vom 2716 (2 KByte) bis zum modernen 27011 (1 MBit). Aber nicht nur EPROMs, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren.
Zum Betrieb benötigt der Junior Prommer nur +5 Volt, die am Joystick-Port Ihres ATARI ST abgenommen werden, alle anderen Spannungen erzeugt die Elektronik des Junior Prommers. Die sehr komfortable Software, natürlich voll GEM unterstützt, erlaubt alle nur denkbaren Manipulationen.
Selbstverständlich läßt sich ein 16-Bit Word in ein High- und Low-Byte zerlegen. Fünf Programmieralgorithmen sorgen bei jedem EPROM-Typ für hohe Datensicherheit. Im eingebauten Hex/ASCII-Monitor läßt sich der Inhalt eines EPROMs blitzschnell durchsuchen oder auch ändern.

Alles dabei!

Bemerkenswert ist der Lieferumfang, so wird z.B. das Fertiggerät komplett aufgebaut und geprüft im Gehäuse mit allen Kabeln anschlußfertig geliefert. Auf der Diskette mit der Treibersoftware befinden sich noch RAM-Disk und ein Programm zum Erstellen von EPROM-Karten, ferner wird der Source-Code für Lese- bzw. Programmierroutinen mitgeliefert und last but not least ist im Bedienungshandbuch der Schaltplan abgedruckt.



Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Hiermit bestelle ich:

Name: ____________
Vorname: ____________
Straße: ____________
Ort: ____________
Unterschrift: ____________

☐ Junior Prommer (Fertiggerät wie beschrieben) nur	DM 189,00	
☐ Leerplatine und Software (o. Bauteile)	DM 49,00	
☐ Leergehäuse (gebohrt und bedruckt)	DM 39,90	
☐ ROM-Karte 128 KByte bietet maximal 4 EPROMS Platz (fertigbestückt o. EPROMs)	DM 58,00	
☐ Easytizer (Fertiggerät)	DM 289,00	☐ Vorauskasse
☐ Easytizer (Teilsatz wie oben beschrieben)	DM 129,00	☐ Nachnahme

Versandkosten: DM 7,50
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr.

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3

Videokamera ein. Dazu sollte die Vorlage möglichst flach sein. Die eingestellte Schärfe ist sehr wichtig und sollte daher wirklich 100%-ig sein! Beobachten Sie auch hier wieder das Bild auf dem Kontrollmonitor.

Für eine Schwarzweiß-Digitalisierung überlassen Sie das Einstellen der optimalen Helligkeit am besten der eingebauten Automatik. Bei Echtfarbdigitalisierung mit Farbscheiben muß die Automatik jedoch unbedingt ausgeschaltet sein. Sie müssen die Helligkeit hier für jede Bildvorlage einmal von Hand einstellen und zwar so, daß das Bild ohne Farbscheiben gerade kurz vor der Überbelichtung steht! Bei der anschließenden Digitalisierung mit den einzelnen Farbscheiben darf die Belichtung auf keinen Fall mehr verändert werden.

Bild 8: Richtige Beleuchtung ist wichtig

PUNKT 3.
WIE WIRD DIGITALISIERT?

Abhängig davon, ob Ihr Digitizer empfindlich gegenüber Störungen durch PAL-Farbsignale ist und Sie einen Videorecorder mit Schwarzweißbetrieb besitzen, nehmen Sie jetzt die Digitalisierung der Vorlagen direkt von der Videokamera aus vor oder zeichnen sie erst auf Videoband auf und spielen sie dann wieder ab. Wenn Ihr Recorder dabei ein wirklich gutes Standbild liefert, genügt eine kurze Aufzeichnung. Sonst nehmen Sie die Vorlage etwas länger auf, als der spätere Digitalisiervorgang Ihres Digitizers dauert. Stellen Sie den Regler *Sharpness* (Schärfe) an Ihrem Videorecorder auf ca. 75%, dann erhalten Sie bessere Ergebnisse.



Bild 9: Das Bild vom Kontrollmonitor

Für eine Echtfarbdigitalisierung mit Farbscheiben oder *RGB-Splitter* benötigen Sie *drei* Digitalisiervorgänge je Bild, je einen für *ROT*, *GRÜN* und *BLAU*. Zeichnen Sie zuerst das Bild ohne Farbfilterung und dann je einmal für jeden Farbwert auf. Achten Sie unbedingt auf die korrekte Reihenfolge der Farben bei der Aufzeichnung und Digitalisierung, sonst erhalten Sie ungewollte Falschfarbenbilder.

Stellen Sie jetzt Helligkeit und Kontrast am Videodigitizer so ein, daß das Ergebnisbild der Digitalisierung möglichst alle Werte von Schwarz nach Weiß besitzt. Achten Sie insbesondere auf eine korrekte Kontrasteinstellung. Ein zu grober Kontrast läßt die Bilder unnatürlich erscheinen, auch wenn Sie auf den ersten Blick schärfer aussehen. Am besten werden die Ergebnisse, wenn das digitalisierte Bild dem Schwarzweißbild des Fernsehmonitors möglichst entspricht. Drehen Sie dazu wenn nötig, die Farbe am Monitor zurück.

Bei der Echtfarbdigitalisierung stellen Sie die Werte anhand des Bildes ohne Farbscheiben ein und lassen die eingestellten Werte für jeden Digitalisierungsvorgang *R-G-B* gleich.

Bedienen Sie den Videodigitizer entsprechend den Angaben des Herstellers. Sind die Ergebnisse bei der Echtfarbdigitalisierung zu blaß, erhöhen Sie den Kontrast und beginnen von vorne. Bei falschen Farben prüfen Sie Weißabgleich und Beleuchtung, eventuell ist Ihnen dann auch ein Fehler bei der Bedienung unterlaufen, oder der eingestellte Kontrast war zu hoch.

Speichern Sie digitalisierte Einzelbilder möglichst so ab, daß Sie den Bildern Namen geben, die dem Bildinhalt entsprechen. Dann brauchen Sie später nicht so lange danach zu suchen. Für eine ganze Bildsequenz eines Computerfilm sollten Sie am besten Namen mit numerisch aufsteigender Endung vergeben, dann können diese Bilder in IMAGIC komfortabel auf einmal eingelesen werden.

Ein erster Einblick

Soviel also zum zweiten Teil unserer Serie BILDWERKSTATT ATARI ST. Wir hoffen, wir konnten Ihnen einen kleinen Einblick in die Technik der Videodigitalisierung vermitteln. Falls Sie an weitergehenden Informationen interessiert sein sollten, insbesondere, was die technische Seite der Videodigitalisierung anbelangt, empfehlen wir Ihnen den Artikel

"PUZZLE - Videodigitizer im Selbstbau"

aus dieser Zeitschrift vom Oktober 1987. Dort werden Aufbau und Funktionsweise des Videodigitizers *EASYTIZER* genau beschrieben.

Zum Schluß möchten wir Ihnen noch ein paar digitalisierte Bilder von den Public-Domain-Disketten BILDWERKSTATT ATARI ST vorführen, die Sie parallel zu dieser Serie erhalten können. So können Sie selbst einen kleinen Einblick in die Qualität digitalisierter Bilder erhalten ...



Bild 10: ... und das digitalisierte Ergebnis

Dazu ein wichtiger Hinweis, den wir in der letzten Folge leider vergessen haben zu erwähnen: Die IMAGIC PD-Version von DENISE läuft wie das Original nur auf Rechnern mit mindestens 1 Megabyte RAM. Wir möchten also die Besitzer von 520 STM und 260 ST-Rechnern, die ihren Rechner nicht aufgerüstet und die Disketten bestellt haben bitten, zu einem Freund oder Bekannten zu gehen, der einen Rechner mit genügend großem Speicher hat, und sich die Computeranimationen dort anzusehen. Es war keine böse Absicht von uns, das zu verschweigen, wir arbeiten nur seit so langer Zeit wie selbstverständlich mit 1 Megabyte-Speicher, daß wir nicht mehr daran gedacht haben, daß es auch noch Rechner mit weniger Speicher gibt.

Für diejenigen unter Ihnen, die die Disketten bereits erhalten haben, folgt jetzt endlich nach der ganzen Theorie eine erste Demonstration (oder haben Sie etwa schon vorher ausprobiert, was sich auf den Disketten befindet?).

Wir empfehlen Ihnen zuvor, eine Sicherheitskopie von den Originaldisketten anzufertigen. Weil wir sehr viele Daten für diese Computeranimation benötigen, haben wir die Disketten in einem Spezialformat erstellt. Dazu wurden je Track 10 Sektoren formatiert. Unter TOS können Sie so eine Diskette nicht komplett kopieren, es gibt jedoch eine ganze Reihe von Kopierprogrammen, die das problemlos erledigen. Wenn Sie kein solches Kopierprogramm zur Verfügung haben, formatieren Sie bitte zwei leere Disketten doppelseitig unter TOS, und kopieren Sie die Disketten dateiweise.

So, nachdem Sie die Sicherheitskopien erstellt haben, starten Sie bitte das Programm *DENISDEM.PRG* von Diskette 1 in Laufwerk A. Das Programm DENISE arbeitet in allen drei Auflösungen des ATARI ST. Wenn Sie in Farbe arbeiten, sollten Sie das Programm jedoch in der niederen Auflösung starten. Einerseits haben wir alle Grafiken auf der Diskette 2 für die niedere und hohe Auflösung erstellt, andererseits haben wir aus Platzgründen das IMAGIC-Runtime-Modul für mittlere Auflösung nicht auf die Disketten kopiert.

Besitzer eines ATARI ST mit einem Megabyte RAM sollten vorher darauf achten, keine RAM-DISK und möglichst wenige ACCESSORIES aktiviert zu haben. Der vollständige Computerfilm benötigt fast den gesamten freien Speicherplatz.

Solange das Programm geladen wird, möchten wir Ihnen ein wenig über DENISE erzählen: Es handelt sich um ein kombiniertes Zeichen- und Animationsprogramm. Sie haben dabei beide Funk-



Bild 11: Kontrast ... zu gering

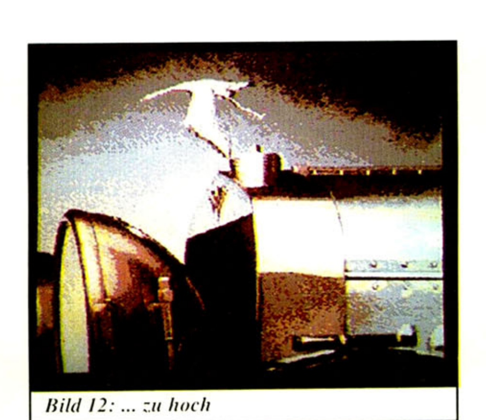


Bild 12: ... zu hoch

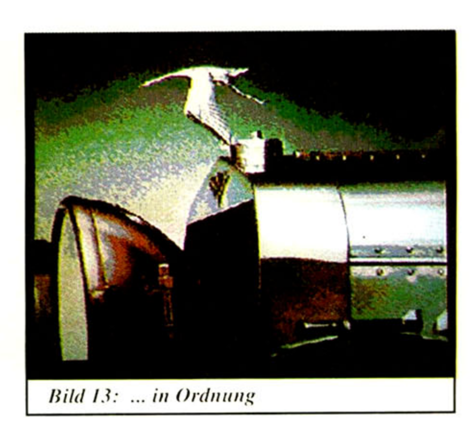


Bild 13: ... in Ordnung

tionen in einem Programm vereint und benötigen normalerweise kein weiteres mehr, wenn Sie einen Computerfilm zusammenstellen wollen. Auch können Sie auch Grafiken aus anderen Programmen übernehmen, solange diese Ihre Bilder in einem der zehn Standardformaten abspeichern, die DENISE einzulesen vermag. Ein Animationsprogramm muß natürlich möglichst viele Bilder gleichzeitig bearbeiten können. Deshalb ist DENISE in der Lage, sehr viele - bei genügend freiem Speicher bis zu 1000 - Bilder gleichzeitig im Speicher zu halten. Und obwohl das Programm sehr umfangreich ist, können Sie in den meisten Fällen weit mehr Bilder im Speicher halten, als dies mit anderen, reinen Zeichenprogrammen möglich ist, weil DENISE fast alle Bilder komprimiert im Speicher hält.



Neben den normalen Zeichenfunktionen enthält das Programm einen Animationsteil, der Filmeditor genannt wird. Wie Sie dort auf einfache Weise einen Computerfilm zusammenstellen können, werden wir Ihnen im Lauf dieser Serie noch zeigen.

Jetzt müßte das Programm DENISE eigentlich geladen sein. Quittieren Sie die Anfangsmeldung mit einem Druck auf die Taste <RETURN>, daraufhin wird das IMAGIC-System installiert. IMAGIC ist ein Subsystem für Computergrafik und stellt viele Grundfunktionen der Bildverarbeitung zur Verfügung, die von DENISE genutzt werden.

Sie sollten sich jetzt in der "DENISE BILD-DATENBANK" befinden. Hier werden alle im Programm befindlichen Grafiken übersichtlich verkleinert dargestellt. Jetzt allerdings ist die Bilddatenbank noch leer. Klicken Sie einmal mit der linken Maustaste auf das Feld *"Ausgang"* ganz rechts unten. Sie kommen damit auf das "DENISE DESKTOP". Legen Sie jetzt die *zweite* IMAGIC-PD-Diskette in das Laufwerk A. Unter dem Menüpunkt *"Grafik"* finden Sie einen Eintrag *"STR laden"*: Klicken Sie diesen Eintrag mit der Maus an. Es öffnet sich eine Fileselector-Box, in der Sie einige Dateien mit der Endung *"*.STR"* sehen. Wenn Sie mit einem Farbmonitor arbeiten, geben Sie jetzt bitte den Dateinamen *"C_FILM.STR"* über die Tastatur ein und drücken die <RETURN>-Taste. Bei einem Monochrom-Monitor geben Sie entsprechend *"M_FILM.STR"* ein.

Jetzt sollte das Programm DENISE beginnen, eine gesamte IMAGIC-Filmstruktur in den Speicher zu laden. Das kann insbesondere im Monochrom-Modus einige Zeit dauern, da hier viele Bilder automatisch von Farbe nach Schwarzweiß konvertiert werden.

Nach erfolgreichem Laden aller Einzelbilder kehrt DENISE automatisch in die Bilddatenbank zurück. Legen Sie jetzt wieder die Diskette 1, von der aus Sie DENISE gestartet haben, in das Laufwerk A ein. Klicken Sie sodann mit der Maus auf das Symbol *"Filmkamera"*. Jetzt sind Sie im DENISE-Filmeditor.

So, zum Abschluß klicken Sie jetzt auf das Symbol *"START"* (laufendes Pferd). Der eingelesene IMAGIC-Film wird gestartet und läuft selbständig ab. Sie können den Film mit Druck auf die Taste <HELP> unterbrechen.

Wie der Film erstellt wurde, zeigen wir Ihnen dann im nächsten Teil unserer Serie.

Also bis zum nächsten Mal ...

Alexander Beller & Jörg Drücker

Die digitalisierten Bilder wurden mit folgender Hardwarekonfiguration erstellt:

Videokamera:	*Philips VHS VIDEO 400*
Videorecorder:	*JVC HR-D725EG*
	baugleich mit DUAL VR 97
Reprostativ:	*Kaiser RS 1*
Digitizer:	*Print & Technik PRO 87, mit **selbstgeschriebener** Digitizersoftware*

ENDE

RELAX

Hallo Relaxer,

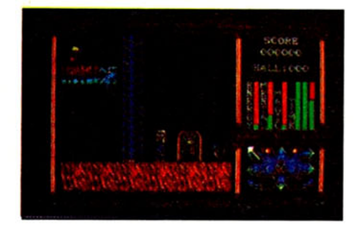
bald kommt der Weihnachtsmann! Sollen auf Ihrem Wunschzettel auch ein paar Computerspiele stehen? Ja, aber Sie wissen noch nicht welche? Na, dann sind Sie bei Relax genau an der richtigen Adresse. Wir möchten Sie auch im Monat des Christkinds in puncto Spielen auf dem laufenden halten. Fünf Spiele im knallharten Test. Außerdem gibt es wieder Kurzmeldungen und einen Messebericht von der Londoner PCS. Nicht schlecht, was? Aber ich rede schon wieder zuviel. Sie wollen ja schließlich etwas über die neuen Games erfahren. Also viel Spaß dabei! Zum Abschluß meines Vorwortes wünsche ich Ihnen noch eine frohe Weihnacht und einen guten Rutsch ins neue Jahr (ist natürlich nicht wörtlich zu nehmen!).Ich würde mich freuen, wenn ich Sie im nächsten Jahr nach dem Streß der Feiertage wieder bei Relax begrüßen dürfte.

Bis bald!

Ihr Carsten Borgmeier

NEWS

"Down at the Trolls" von Rainbow Arts ist nun endlich fertig. Seit Monaten macht Rainbow Arts uns mit Anzeigen auf das neue Kletter-, Such- und Sammelspiel heiß. Doch als ich das neue Game



gesehen habe, wurde mir wieder ganz kalt: Das ruckelige Scrolling verdirbt einem den ganzen Spielspaß. Das kann auch der Editor zur Konstruktion von eigenen Schatzsucherhöhlen nicht wieder gutmachen. Einziger Lichtblick ist die lustige Animation der Trolle, die versuchen, mit Knüppeln auf den Helden einzuschlagen.

Info: Rushware

Das französische Softwarehaus Titus kommt nach **"Crazy Cars"** und **"Fire & Forget"** mit einem neuen Spielchen auf den Markt. Mit einem Rennboot rasen Sie über verschiedene Rennkurse, überholen die Boote der Kontrahenten und schlafen vor Langeweile ein. Das Spiel ist viel zu leicht und bietet zudem noch wenig Abwechslung.

Info: News Klug

"Infogrames" hat zwei neue Titel für den harten Mann. In **"Operation Jupiter"** soll der Spieler eine von Terroristen besetzte Botschaft stürmen



und Geiseln befreien. Grafik und Sound sind vom Feinsten. Nur ist das Game für meine Begriffe streckenweise ein wenig zu brutal. In **"Action Service"** werden stählerne Soldaten für lebensgefährliche Einsätze ausgebildet. Auf einem Hindernisparcours robbt der angehende Elitekämpfer durch den Matsch, springt über Stacheldraht und marschiert durch Minenfelder. Der Spieler kann den Durchlauf durch den Hindernisparcours mit einer Videooption aufzeichnen und nach Wunsch wieder anschauen. In einem Editor besteht die Möglichkeit, immer kniffligere Hindernisstrecken zu entwickeln.

Info: Bomico

"Bard's Tale", den Klassiker unter den Rollenspielen, gibt es jetzt auch für ATARI ST.

Info: Leisuresoft

Magic Bytes kündigt einen Hammer fürs Weihnachtsgeschäft an: Auf einer Compilation namens **"It's a kind of Magic"** bietet das Gütersloher Softwarehaus die international erfolgreichen Titel **"Pink Panther"**, **"Western Games"**, **"Vampires Empire"** und **"Clever & Smart"** an. Für Sammler eine tolle Gelegenheit!

Info: Ariolasoft

"Football Director II" heißt ein neues Managerprogramm für Fußballfans, das über viele Spielfeatures verfügt. Leider macht das Spiel den Eindruck, als sei es in der Mittagspause programmiert worden. Drückt man längere Zeit auf die Space-Taste, scrollt der Text aus dem Bildschirm. Wenn man schon in Basic programmiert, sollte man wenigstens durch eine gute Eingabemaske dafür sorgen, daß es keiner merkt.

Info: Leisuresoft

Auf vier Disketten ist die Handlung von **"Chrono Quest"**, dem neuestem Adventure von "Psygnosis", verteilt. Herrliche Grafiken und eine interessante Handlung, bei der eine Zeitmaschine im Mittelpunkt steht, machen das Game spielenswert. Ursprünglich wurde der Titel unter dem Namen "Explora" angekündigt. "Psygnosis" hat vom französischen Hersteller die Lizenz gekauft, eine Speichermöglichkeit des Spielstandes eingebaut, und fertig war das neue CHRONO QUEST. Für Adventure-Freunde ein absolutes Muß.

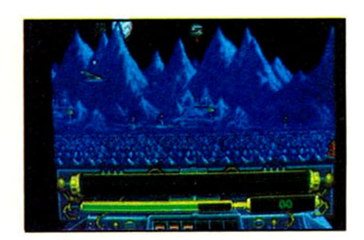
Info: News Klug, Leisuresoft

Die Firma "Digital Artwork", in Heft 6/88 als "Ein neuer Stern im Softwareland" vorgestellt, hat sich aus der Computerspieleindustrie zurückgezogen. Die talentierten Grafiker und Programmierer von Digital Artwork kehrten DA-Geschäftsführer Jens Müller den Rücken und gründeten mit Thalion Software ihre eigene Firma. Thalion Software kündigt das Rollenspiel **"Dragonflight"** (siehe Relax 6/88) für den Februar 1989 an.

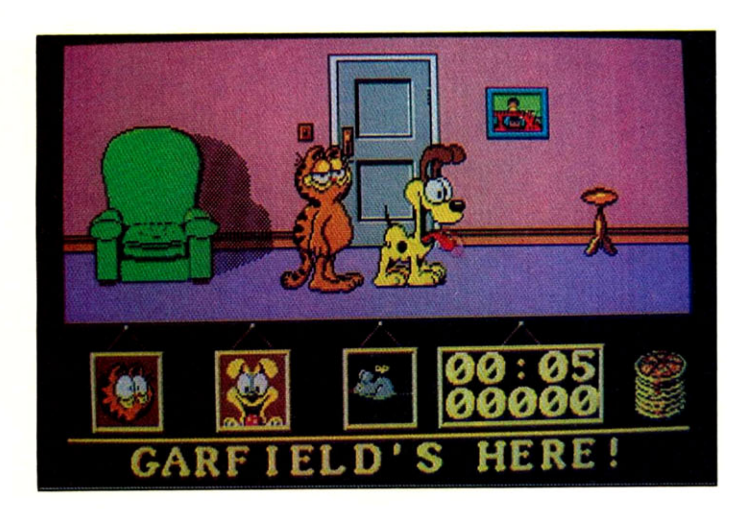
Info: Ariolasoft

"Logotron" hat **"Starray"**, eine neue Defendervariante mit rasend schnellem, horizontalem Scrolling (soll angeblich auf dem ST nicht möglich sein!!) entwickelt.

Info: News Klug



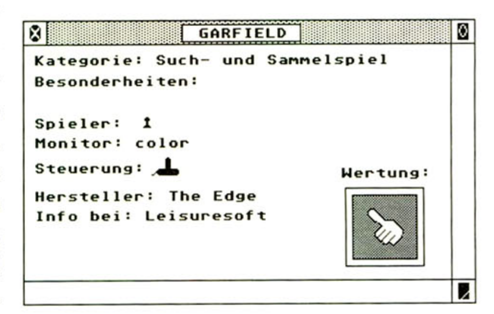
Garfield



Wer kennt ihn nicht, den fetten, frechen Nimmersatt? Die Rede ist von U.S-Comic-Star Garfield. Der orange-gelbe Kater gewinnt auch bei uns immer mehr Freunde. Es gibt ihn auf T-Shirts, Aufklebern, Hörspielcassetten und jetzt auch für den ATARI ST. "The Edge" zeichnet für das Game verantwortlich. In "Garfield - big, fat, hairy deal" geht es um folgendes: Garfield`s Freundin Arlene ist von Tierfängern ins städtische Tierasyl geschleppt worden. Garfield und Odie machen sich sofort auf den Weg, um das Katzenmädchen zu befreien. Sie steuern Garfield, der sich in vier Richtungen bewegen kann, durch die Stadt. Per Joystick geben Sie Garfield zu verstehen, was er machen soll:

Der Comic-Kater kann springen, Gegenstände aufnehmen und ablegen, essen oder dem dummen Odie einen Tritt in den Allerwertesten geben. Die Animation hierbei ist hervorragend gelungen. Die Programmierer haben den Comic-Star gut getroffen: Sein hämisches Grinsen, der müde Gang, die Freude, wenn Odie getreten worden ist - alles wurde herrlich animiert. Sie steuern Garfield durch mehrere Screens, zwischen denen hin- und hergeschaltet wird. Von Scrolling haben die Programmierer wohl nie etwas gehört? Unterwegs finden Sie nützliche Gegenstände wie Taschenlampen, Knochen, Hamburger usw., die Sie einsammeln können. In einem Fenster am unteren Bildschirmrand zeigt der ST die gesammelten Gegenstände an. Ferner sind eine Uhr und eine "Pizza"-Anzeige zu sehen. Sind alle Pizzen verschwunden, fällt Garfield kopfüber in einen tiefen Schlaf. Im Spiel sind einige lustige Gags versteckt. So bekommt Garfield beispielsweise irgendwann eine Torte an den Kopf geworfen. Ohne diese Späße, die gute Grafik und den flotten Sound wäre das Game ein langweiliges Such- und Sammelspiel. Dank der lustigen Aufmachung aber verschönert "Garfield" viele Spielstunden.

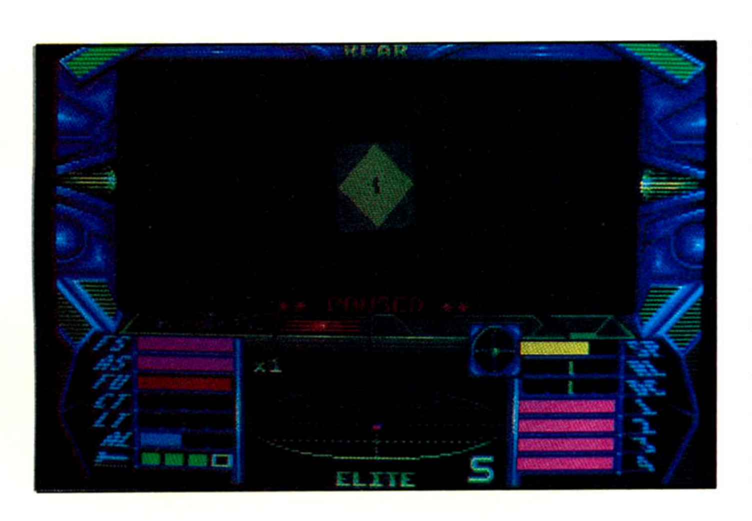
CBO


GARFIELD

Kategorie: Such- und Sammelspiel
Besonderheiten:

Spieler: 1
Monitor: color
Steuerung:
Wertung:
Hersteller: The Edge
Info bei: Leisuresoft

ELITE



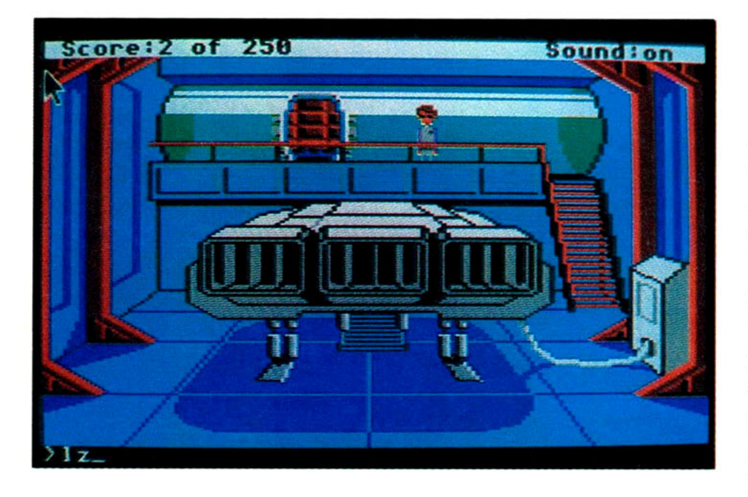
Spielefans der 8 Bit-Rechner kennen es schon: ELITE, das komplexe Action- und Strategiespiel, zu dessen Lösung es viel Zeit und Geduld bedarf. An Bord des Kampfraumschiffes Cobra Mk II dringen Sie in die unendlichen Weiten des Weltraums ein. Ziel des Spiels ist es, den Rang ELITE zu verdienen. Doch bis dahin ist es ein langer Weg. Am Anfang des Spiels bekleiden Sie den Rang HARMLESS (harmlos), sind nicht vorbestraft, haben ein geringes Kapital (100 Credits), einen leeren Laderaum und ein schwach bewaffnetes Schiff. All diese Eigenschaften verändern sich während des Spielverlaufes, je nachdem wie geschickt oder ungeschickt Sie agieren. Zu Beginn befindet sich die Cobra Mk II im Hangar einer Raumstation. Hier bereits nehmen Sie die Geschäftstätigkeit auf. Per Knopfdruck kommt ein Bild zum Vorschein, auf dem Handelswaren wie Edelmetalle, Nahrungs-

mittel, radioaktive Stoffe usw. in Form von Piktogrammen zu sehen sind. Mit dem Joystick wird die entsprechende Ware ausgewählt. Dabei verringert sich selbstverständlich das Kapital und das verfügbare Ladevolumen. Eine andere Taste führt Sie ins Waffendepot. Hier werden - ebenfalls für Geld (denn auch im Weltraum gilt: Ohne Moos nix los!) - Waffen wie Zusatzlaser, Energiebomben, und Raketen am Schiff montiert. Ist das Schiff komplett ausgerüstet, wird es auf Knopfdruck aus der Station katapultiert. Im Weltall besteht dann die Möglichkeit, nach allen vier Himmelsrichtungen zu blicken. Unterhalb des Cockpitfensters befindet sich eine Anzeige, die über Tankinhalt, Energie, Temperatur des Lasers, Geschwindigkeit, Raketenanzahl und Schutzschildstärke informiert. In der Mitte ist ein dreidimensionaler Radarschirm installiert. Behalten Sie den Schirm gut im Auge! Während des Fluges passiert nämlich eine ganze Menge: Piratenschiffe greifen an. Wenn die Cobra nicht ausreichend bewaffnet ist, sollte man sich in einem solchen Falle schnellstens verdrücken. Die Piraten neigen nämlich dazu, die gesamte Ladung zu stibitzen. Piratenschiffe, aber auch andere Handelsschiffe und Himmelskörper, sind in schneller, ausgefüllter Vektorgrafik zu sehen. Es lohnt sich, von Zeit zu Zeit einige dieser Schiffe abzuschießen, da in den Trümmern wertvolle Waren oder Waffen herumschwirren, die Sie kostenlos aufsammeln dürfen. Dummerweise können Ihnen nicht nur Piratenschiffe gefährlich werden. Wenn Sie sich selber als Pirat versuchen, fühlen sich Kapitäne anderer Handelsschiffe bedroht und eröffnen das Feuer auf die Cobra. Mit friedlichen Absichten im Sinn und dem Vorsatz, Geld auf legale Weise zu verdienen, blendet der ST auf Wunsch eine Grafik mit der momentanen Galaxie ein. Auf dieser Karte sind alle in diesem Teil des Universums existierenden Planeten vermerkt. Auf Knopfdruck fliegen Sie mit Überlichtgeschwindigkeit zum gewählten Ziel. Hier gilt es, eine Raumstation ausfindig zu machen, auf der Sie das Schiff landen können. Fingerspitzengefühl wird bei der Landung verlangt, denn die Raumstation dreht sich im All. Ist das Schiff sicher durch die Öffnung geflogen, sollten die im Laderaum befindlichen Waren verkauft werden. Über den auf der Raumstation herrschenden Marktpreis informiert eine weitere Tafel, die auch schon während des Fluges abrufbar ist. Ein kleiner Tip: Kaufen Sie sich von dem erhaltenen Geld erst Benzin, dann Waffen und Handelswaren und zum Schluß nützliche Dinge wie Andockautomatiken, Rettungskapseln usw. Je nachdem wieviel Geld Sie gemacht haben und wie sich Ihr Ruf (Piraterie, Raumschlachten,...) entwickelt hat, klettern Sie in der Rangfolge höher. Doch zum Endziel ELITE bedarf es einiger Monate. Passen Sie nämlich nicht auf, wird Ihr Schiff in Staub verwandelt und das Spiel beginnt von vorne. Aber keine Panik: Rangstatus, Bewaffnung und Geldvorrat sind abspeicherbar, so daß an alter Stelle weitergespielt werden kann. Abschließend noch etwas zu Grafik und Sound: Schnelle und ausgefüllte Vektorgrafik verzückt das Auge. Sound kommt während des Spiels so gut wie gar nicht vor. "Elite" hat diese Effekthascherei aber auch nicht nötig; die Schwerpunkte liegen beim strategischen Anteil. Fans von Strategiespielen können sich auf viele unterhaltsame Spielstunden im Weltraum freuen. "Elite" ist sein Geld echt wert.

CBO



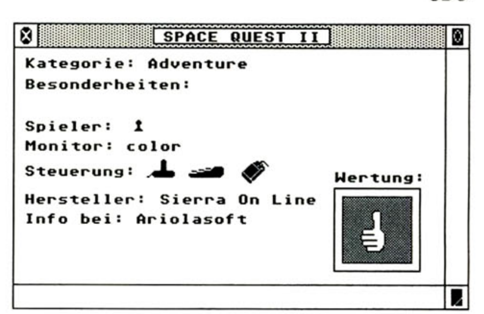
Space Quest II - Die Rache



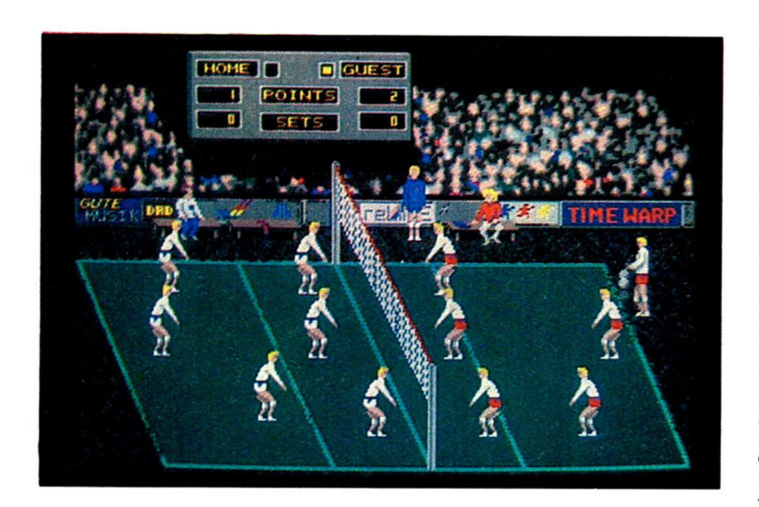
Sludge Vohaul, der wahnsinnige Wissenschaftler, der schon in Space Quest I für Aufregung sorgte, ist wieder aus den unendlichen Weiten des Weltraums zurückgekehrt, um das Sternensystem zu tyrannisieren. Er will seinen Plan in die Tat umsetzen, den Planeten Xenon zu vernichten. Ein sadistischer Plan, erdacht von einem kranken Gehirn. Sie, als Roger Wilco, Techniker und guter Kumpel, sind bereit, den teuflischen Plan zu verhindern. So präsentiert sich die Vorgeschichte von "Space Quest II". Hersteller Sierra on Line, bekannt durch große Erfolge wie "Police Quest", "Leisure Suit Larry" und natürlich auch "Space Quest" hat ein weiteres Spitzenadventure des Weltraums fertiggestellt. Das Game beginnt damit, daß Roger Wilco gerade mit einem Besen die Raumstation von außen säubert. Als ihm der Schrubber entgleitet und in den Tiefen des Alls verschwindet, übernehmen Sie die Steuerung des Helden. Wie bei Sierra Adventures üblich, geschieht das per Joystick und Tastatur. Mittels des Knüppels lenken Sie Roger in alle Himmelsrichtungen. Mit Tastaturkommandos geben Sie ihm zu verstehen, was er tun soll. Der Parser ist exzellent und versteht einzelne Wörter genauso wie ganze Sätze. Das Adventure fesselt stundenlang an den Bildschirm. Bei "Space Quest II" hatte ich mal wieder Schwierigkeiten, mich vom ST loszueisen und meine Testberichte zu schreiben. Als mir der Besen entglitt, ging ich zur Luftschleuse, um mich nach

einem neuen umzusehen. Da erblickte ich einen Raumanzug, den ich anzog. Ein Kollege stolperte auf mich zu und verlangte, ich solle mal einen Blick in die Raumfähre riskieren. Folgsam lief ich in Richtung Raumfähre. Plötzlich wurde ich zusammengeschlagen und verlor das Bewußtsein. Als ich mich von den Schlägen erholt hatte und wieder aufwachte, teilte mir jemand mit, daß man gedenke, mich auf den Dschungelplaneten zu transportieren. Meine Wachen bringen mich, die Strahlenkanonen im Anschlag, auf eine Schwebeplatte. Ich dachte schon, jetzt ist es um mein Heldenleben geschehen. Da passierte etwas Unvorhergesehenes: Die Untertasse fing an zu stottern und stürzte ab. Glücklicherweise hatte ich beim Aufprall eine weiche Landung. Ich plumpste auf einen meiner Bewacher. Die beiden Wächter waren mausetot. Ich sah mich um, wo ich gelandet war. Ringsum ein riesiger Dschungel. Oh nein! Ich dachte: "Mal sehen, ob es irgendwelche Gegenstände im Wrack oder bei den toten Wächtern gibt, die mir weiterhelfen." Nichts! Also, machte ich mich auf den Weg durch den Dschungel. "Vielleicht finde ich ja eine Stadt. Da gibt's bestimmt Telefon. Dann rufe ich meinen Chefredakteur an, der holt mich ab, und ich bin aus aller Misere." Doch von einer Stadt war weit und breit nichts zu sehen. Stattdessen fand ich einen merkwürdig aussehenden Pilz, den ich näher betrachtete. Doch als ich dem Pilz näher kam, verschlang mich dieser sogleich und setzte ein zufriedenes Grinsen auf. Für mich war das Abenteuer vorbei. Wenn ich Ihnen jetzt noch sage, daß ich bei meinem ersten Testspiel nur 7 von 250 möglichen Punkten erreicht habe, können Sie sich vorstellen, wieviele Rätsel Ihnen das Spiel aufgibt. Grafisch ist es wie alle Sierra Adventures eher schlicht ausgefallen. Dafür gibt es aber unzählige Räume und Personen, die Sierra insgesamt auf drei Disketten verteilt hat. Ich für meinen Teil hatte viel Spaß mit "Space Quest II". Gehen Sie doch mal in Ihren Computershop und schauen Sie sich das Game an. Es wird Ihnen bestimmt gefallen.

CBO



Volleyball-Simulator

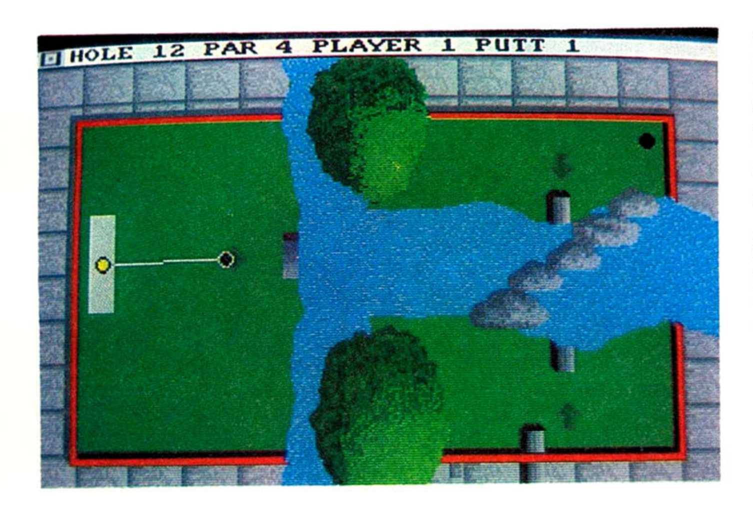


In den Sporthallen und auf den Fernsehbildschirmen erfreut sich in den letzten Jahren eine Sportart immer größerer Popularität: Volleyball. Dank Time Warp Productions hält diese beliebte Mannschaftssportart auch Einzug auf dem ST. Der "Volleyball-Simulator" bringt Abwechslung in die Sportspielszene. Er überzeugt durch ein eigenständiges Konzept, das sich an keinem Vorgängerprogramm orientiert. Begeistert war ich auch von der nahezu perfekten programmiertechnischen Umsetzung auf dem ST und der Realitätsnähe des Spiels. Alle Features, die das Volleyballspiel in der Sporthalle ausmachen, sind auch im Volleyball-Simulator zu finden. Die 12 Spieler auf dem Feld können pritschen, baggern, hechtbaggern, schmettern, blocken und natürlich auch einen Lop durchführen. Gespielt wird nach den original Volleyballregeln, entweder alleine oder mit einem Partner gegen den Computer. Es besteht auch die Möglichkeit, gegen einen menschlichen Gegner eine Partie Volleyball zu wagen. In jedem Match kontrolliert der Spieler einen Volleyballspieler, die Steuerung der anderen fünf übernimmt der Computer. Da kommen aufregende Ballwechsel zustande, der Computergegner spielt nämlich ziemlich stark. In diesem Programmteil kommen die Actionfans voll auf ihre Kosten. Doch "Time Warp" hat auch für Strategiefans und Taktiker etwas zu bieten. Mit "Escape" gelangt man in den Taktikeditor, in dem man neue Spielzüge und Spielformationen für seine Mannschaft bestimmt. Diese Spieltaktiken können dann abgespeichert und vor dem nächsten Match in den Speicher geladen werden. Ihr Team spielt dann nach der ausgearbeiteten Taktik.
Eine tolle Idee! Volleyball-Simulator ist ein echtes Spitzenspiel, das nicht nur über eine neue, originelle Spielidee verfügt, sondern auch hervorragend umgesetzt wurde. Die Animation der Sportler ist bei allen Aktionen sehr flüssig, wenn auch manchmal ziemlich langsam. Die Soundeffekte während des Spiels und vor allem der Titelsound verdienen Anerkennung. Wenn der ST das Titelbild zeigt, spielt er eine fetzige, digitalisierte Melodie, die in einem Tonstudio eigens für das Spiel komponiert wurde. Schade nur, daß man den Monitor auf volle Lautstärke stellen muß, um etwas mitzubekommen. Trotz kleiner Mängel bei Animation und Sound ein Spitzentitel.

CBO

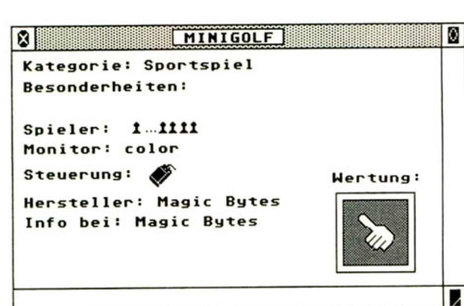


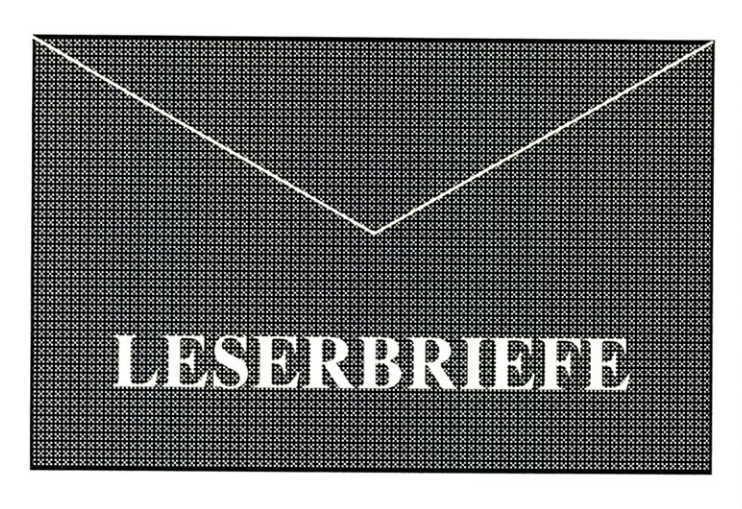
Minigolf



Wenn im Frühling die ersten Sonnenstrahlen durchkommen, dann öffnen die zahlreichen Minigolfanlagen in der Bundesrepublik wieder ihre Tore und geben Tausenden Gelegenheit, sich mit Schläger und Ball an den 18 Minigolfbahnen zu messen. Doch bis dahin müssen noch einige Monate vergehen. Das Gütersloher Softwarehaus Magic Bytes will allen Minigolffans die Wartezeit mit einem tollen Programm versüßen. Jetzt haben die ST-User auch Gelegenheit, in den eigenen vier Wänden zu golfen. Dazu muß nicht das Wohnzimmer umgeräumt werden. Ein ATARI ST mit Maus und "Minigolf" von Magic Bytes reichen vollkommen aus. Das Spielgeschehen ist aus der Vogelperspektive dargestellt. Zu Beginn jeder neuen Golfbahn plaziert man die Kugel an der Abschlagstelle. Dann heißt es, eine Linie zu dem Punkt zu bewegen, wo die Kugel auftreffen soll. "Minigolf" hat ein wenig Ähnlichkeit mit Billard. Auch hier gilt die Regel aus dem Physikunterricht: "Einfallswinkel= Ausfallswinkel". Auf jeder Bahn befinden sich andere Hindernisse. Von der schlichten rechteckigen Bahn bis hin zum verzwickten Rohrsystem mit kleinen Brücken geht die Variationsvielfalt. Mit etwas Übung lassen sich aber alle Bahnen meistern. Für interessant halte ich einige Optionen, die mit der Maus in einigen Pull Down Menüs aufgerufen werden können. So ist es beispielsweise möglich, sich den letzten Schlag noch einmal anzuschauen oder einen mißglückten Ball nochmals zu schlagen. Sound und Grafik sind bei Minigolf nicht der Rede wert. Aufgrund der Tatsache, daß das Geschehen aus der Vogelperspektive dargestellt ist, macht die Grafik einen sehr schlichten Eindruck. Minigolf begeistert dafür aber durch das einfache und unterhaltsame Spielprinzip. Besonders viel Spaß macht Minigolf mit mehreren Spielern. Bis zu vier Golfer können an einer Minigolforgie teilnehmen. Ich habe mit einigen Freunden eine ganze Nacht lang dieses unterhaltsame Game gespielt: Minigolf ist ein süchtig machendes Golfspiel, das die Maus zum Glühen bringt.

CBO





Zwei GFA-BASIC-Probleme

Erstens:

Bei Anwendung des Befehls EVEN(X) (GFA-BASIC 2.0) erkennt der Computer (520 STM) die Zahl 5734125917 als GERADE Zahl. Wie läßt sich das erklären?

Zweitens:

Der Abbruch eines Programmes durch die Tasten *"Control"*, *"Alternate"* und *"Shift"* läßt bei sich meinem Computer nicht durchführen. Eine Überprüfung der Tastatur (Widerstände, ICs, Dioden) und deren teilweisen Austausch (ICs) brachte nicht den gewünschten Erfolg.

Haben Sie eine Lösung für meine Probleme?

(Kerstin Müller, Tuttlingen)

Red.:

Erstens: Die Tatsache, daß Ihre obengenannte Zahl nach GFA-BASIC gerade sein soll, ließ auch bei uns gewisse Zweifel aufkommen. Bei der Überprüfung mit GFA-BASIC-2.0 stellten wir fest, daß dieser Fehler ab der Zahl 5000000001 auftritt, was wohl sicherlich an der Ungenauigkeit der Rechenroutine liegen dürfte. Bei der Version 3.0 tritt dieser Fehler nicht mehr auf und scheint daher behoben zu sein.

Zweitens: Der Fehler ist schon länger bekannt...und behoben. Zunächst einmal können wir Sie beruhigen: An ihrer Tastatur ist nichts kaputt, sondern der Fehler liegt in der Abfrage der *Control-Shift-Alternate*-Tasten: In der alten TOS-(Betriebssystem)-Version vor dem Blitter-TOS konnte man diese Tasten nur über eine nicht offiziell freigegebene Adresse erfragen, so daß einige Programme, darunter auch GFA-BASIC-2.0, diese *'unsauber'* abgefragt haben. Es kam wie es kommen mußte, denn ab dem Blitter-TOS stimmte diese Adresse nicht mehr. (Nur am Rande für alle Programmierer; heute kann man diese Adresse ganz offiziell erfragen, denn das funktioniert wie folgt: Mann holt aus der (dokumentierten) Adresse $4f2 mit Namen *_sysbase* den Zeiger auf den Anfang der Betriebssystemheaders, der an der 36. Stelle einen Zeiger auf *kbshift* enthält - in älteren Betriebssystemversionen ist *kbshift* an der Adresse $1eb zu finden). Ihr zweites Problem läßt sich genauso lösen wie das erste und zwar, in dem Sie sich entweder ein Update von GFA-BASIC-2.0 oder am besten gleich ein Upgrade auf Version 3.0 besorgen.

*

50 Hertz contra 60 Hertz

Ich besitze einen 260 ST und den *"Thomson for ATARI"* (CM 36382 AR)-Farbmonitor, der mit der Rasterfrequenz 50 Hz läuft. Bis vor kurzem benutzte ich das Disketten-TOS (RAM-TOS) und hatte damit auch keinerlei Schwierigkeiten. Vor einigen Wochen lies ich mir nun das Blitter-TOS in meinen ST einbauen und mußte leider nach dem Einbauen feststellen, daß der Monitor unerträglich flimmerte. Nachdem ich das Utility-Programm *"Change-Hertz"* aufgerufen hatte, stand das Bild wieder ruhig. An dem Original-ATARI-Farbmonitor eines Bekannten funktionierte mein ST mit Blitter-TOS tadellos.

Da ich nun nach jedem Reset erst dieses Change-Hertz-Programm laden muß, kann ich eine ganze Reihe von Programmen, die mit Reset gestartet werden, nicht nutzen.

Meine Frage lautet daher: Gibt es eine Möglichkeit, den Monitor so an meinen ATARI ST anzupassen, daß ich alle Programme laufen lassen kann?

(Malte Thoma, Freiburg)

Red.: Dem Manne kann geholfen werden: Das Problem liegt darin, daß Ihr Monitor nur auf 50 Hz läuft und der ATARI auf 60 Hz initialisiert wird, was eine Frequenz ist, die Ihr Monitor (der ATARI-Monitor aber doch) leider nicht mehr verarbeitet. Natürlich können wir Ihnen keine Angaben machen, wie Sie Ihren Monitor intern verändern müssen, aber es gibt eine Möglichkeit, den ST ein wenig zu verändern.

ACHTUNG: Durch das Öffnen des STs verlieren Sie Ihre Garantie an dem Gerät, wenden Sie sich also gegebenenfalls an Ihren Händler. Sehr viele ST-Besitzer haben sich ihr Blitter-TOS auf EPROM gebrannt und konnten dadurch Änderungen vornehmen. Interessant an Ihrem Problem ist, daß der ATARI mit dem originalen Blitter-TOS auf 50 Hz initialisiert wird, die Ihrem Monitor nicht zu schaffen machen dürfte. Daraus folgere ich, daß es sich bei Ihnen um ein schon modifiziertes Blitter-TOS in EPROMs handelt (Das TOS Ihres Bekannten ist demnach nicht manipuliert). Aufgrund diese Änderung wird Ihr ATARI auf 60 Hz eingestellt, die Sie (abgesehen von CHANGE-HERTZ) durch Ändern eines Ihrer EPROMs rückgängig machen können. Dazu sollten Sie das Byte an der Adresse FC001D ändern, aus dem beim Start des TOS gelesen wird, ob der PAL- oder NTSC-Mode eingeschaltet werden soll. Dies wird durch das Bit 0 bewirkt, wobei ein gesetztes Bit 50 Hz und ein gelöschtes Bit 60 Hz bedeuten. Demzufolge müßte bei Ihnen dieses Bit gelöscht sein, was Sie mit einem Speicher-Monitor wie zum Beispiel *TEMPELMON* leicht überprüfen können. Wenn Sie sich nun Ihr entsprechendes EPROM auslesen, ändern (Bit 0 setzen, so daß nun eine 3 in der Adresse zu finden ist) und wieder brennen, was zum Beispiel mit dem *Juniorprommer* geschehen könnte, dann dürfte Ihr Problem erledigt sein.

*

Metacomco-Assembler-Probleme

In der ST 8/9-Ausgabe wurde ein Sound-Sampler-Programm vorgestellt, welches mit dem Ideal-Assembler von Omikron geschrieben ist. Da ich im Besitz eines Metacomco-Assemblers bin, versuchte ich das Programm zu übertragen, wobei aber Metacomco keinen einzigen LEA-Befehl wie zum Bespiel Zeile 134 *LEA clrmtext(PC), A0* annimt. Gebe ich diesen Befehl unabhängig von diesem Programm ein, nimmt der Compiler ihn an, aber im Zusammenhang mit dem Sampler-Programm zählt er mir alle LEA-Befehle als Fehler beim Assemblieren auf.

Vielleicht wissen Sie weiter?

(Timo Hamme, Rüsselsheim)

Red.: Nach etwas Nachforschen sind wir auf die Lösung

des Problems gestoßen, so daß wir Ihnen mitteilen können, daß der Fehler nicht am eigentlichen LEA-Befehl liegt, sondern in der für den Metacomco unverständlichen Definition von clrmtext in Zeile 692. Dort steht vor dem *'dc.b'-Pseudoopcode* ein PUNKT, der den Metacomco ins Schleudern bringt. Entfernen Sie also *ALLE* Punkte vor den dc.b-Pseudeopcodes und Sie müßten vor einem weiteren Auftreten dieses Problemes gefeit sein.

*

Tonloser Multisync Monitorbuchsenbelegung

Angeregt durch Testberichte und - wie ich heute weiß - unzureichende Beratung durch meinen Fachhändler in *Norderstedt*, habe ich mir als Quasi-Anfänger zu meinem 1040ST einen *MultisyncII*-Monitor gekauft. Zu meinem Entsetzen ist das Ding aber *"tonlos"*, so daß ich einen großen Teil meiner Programme nicht oder nur bedingt einsetzen kann. Da mein Fachhändler bis zum heutigen Tage mit dem Problem der Tonbeschaffung überfordert scheint.

Meine Frage an Sie:

Die Monitorbuchse des ST
(Draufsicht)

1 - Audi Ausgang
2 - H+V- Sync-Signalmischung
3 - Port A, Bit 6, Soundchip
4 - Monochrom-Sensor
5 - Audi-Eingang
6 - Grün-Signal
7 - Rot-Signal
8 - + 12V out (max 10mA)
9 - Horizontales Sync-Signal
10 - Blau-Signal
11 - Monochromsignal
12 - Vertikales Sync-Signal
13 - Masse

Woher kann ich aus meinen Programmen den Ton (Mono reicht völlig) zaubern und was für Geräte muß ich mir zusätzlich anschaffen?

(Heinz Norbert Loeblein, Norderstedt)

*

Ich möchte meinen Kassettenrekorder an den ATARI ST anschließen, weiß aber nicht wo und wie. Mein Radio besitzt folgende Anschlußstellen: 2 Klinkenkupplungen 3.5 mm (Lautsprecher), 2 Klinkenkupplungen 3.5 (Mikrofon), 1 Klinkenkupplung 5mm (Kopfhörer), 1 Diodenkupplung (Überspielkabel).

(Martin Büttner, Nordheim)

Red: Wie man im Bild erkennen kann, liegt an PIN 1 der Monitorbuchse des ATARI ST das Audio-Signal. Dieses Signal kann über eine Verstärker abgegriffen werden, was zum Beispiel über ein *AUX-* oder *TAPE-Input-Eingang* eines HiFi-Verstärker geschehen kann (Das Anschließen eines Lautsprechers direkt am ST funktioniert nicht). Dazu muß man natürlich an diesen Pin herankommen und einen Adapter bauen, aus dem das Audio-Signal herauskommt und der die anderen Signale an den Monitor weiterleitet. Ein **Tip:** Viele ST-Besitzer kaufen sich im Laufe der Zeit einen sogennanten Monitorumschalter, mit dem man zwischen Schwarzweiß- und Farbmonitor hin- und herschalten kann. Die meisten dieser *Monitorumschalter* haben auch einen Ausgang, an den man die HiFi-Anlage anschließen kann (siehe auch *'Multisync am ST'*).

*

Multisync am ST

Kann man an einen ATARI MEGA ST einen Multisync-Farbmonitor (z.B. Eizo 8060 S) anschließen? Wenn ja, läuft dann auch Software, die einen S/W-Monitor erfordert (z.B. STAD), wenn ich den Farbmonitor auf Textdarstellung umschalte?

(A. Brencher, Brühl)

Red: Sicherlich läßt sich ein Multisync-Monitor an den ST anschließen. Er ist sogar hervorragend geeignet, da er, im Gegensatz zu normalen Farbmonitoren, die Farb- UND die Schwarzweiß-Auflösung des ST darstellen kann. Einen speziellen Monitorumschalter für Multi-Sync-Monitore gibt es bespielsweise von *3K-EDV-Entwicklungen* oder ein Multi-Sync-Kabel bei der Firma *Lindy*. In die Textdarstellung müssen Sie nicht schalten, denn beim ATARI gibt es keine echte Textdarstellung, da alles auf Grafik basiert. Die Monochrom-Darstellung nimmt beim ATARI ST nur deshalb eine Sonderstellung ein, weil Sie mit einer hohen Bildwiederholfrequenz von 70 Hz und der höchsten Auflösung des ST von 640x400 Punkten läuft.

Datobert Business

die logische Weiterentwicklung zum PD-Programm DATOBERT. Zur grafischen Darstellung Ihrer Geschäftsbilanzen, Schulnoten, Haushalts-, Auto-, Heizkosten u.v.m. 15 verschiedene, z.T. dreidimensionale Grafiktypen, wie z.B. Landschafts-, Icon-, oder Picturegrafik. Integrierte Statistikfunktion. Daten aus VIP-Professional können direkt übernommen werden.

DatoFakt

leistungsfähige Fakturierungssoftware mit einer Schnittstelle zu DATOBERT BUSINESS.

DATOBERT BUSINESS	89.00 DM
DATOFAKT	248.00 DM

Handbuch mit Demodisk vorab je DM 20.-(wird bei Kauf angerechnet)

baumann computer

Obere Schwemmbichlstr. 25
8371 Kirchdorf

Golem-Laufwerk 3,5 Zoll	DM 279,00
20-MB Festplatte kpl. anschlußf.	DM 899,00
BTX-Term an Postmodem	DM 249,00
Lattice C V 3.04 Deutsch	DM 249,00
Megamax Modula 2 Deutsch	DM 319,00
STAD Deutsch	DM 149,00
Signum 2 Deutsch	DM 369,00
PC-Ditto MS-Dos Emulator Dt.	DM 159,00
MCC-Makro-Assembler V 11.1	DM 99,95
MCC Pascal V 2.02	DM 175,00

Kostenlose Prospekte,
auch für IBM und Amiga gibt's bei...



CWTG Joachim Tiede
Bergstraße 13 ★ ★ ★ 7109 Roigheim
Tel./BTX 0 62 98/30 98 von 17-19 Uhr

DELO Comp. Tech.

DISKETTENSTATION für **ATARI ST**
D 26 mit NEC 1037 A doppelseitig 3,5" 1 MB
– Test ATARI magazin 9.88
– komplett anschlußfertig **275,–**

dto. mit 2 Floppybuchse D 25	318,–
Doppelstation D 50 2×1 MB mit 2 NEC 1037 A kompl. anschlußfertig	nur 498,–
Speichererweiterung für ST	lieferbar
Vortex HD 20 plus	**1298,–**
Vortex HD 30 Plus	**1498,–**
Vortex HD 60 plus	**2398,–**

NEC FD 1037 A mit Anschlußbelegung 179,–

NEC P6+ ... 1648,–	NEC P6+ color ... 1928,–
NEC color Nachrüstung P6 . 348,–	P6+ . 295,–
NEC Multisync GS ... 535,–	NEC P2200 ... 859,–
Monitorswitchbox für alle ST anschlußfertig	37,–
Druckerswitchbox für ATARI ST	55,–
AMSTRAD Computer	lieferbar
Sybex ST Kontor TOS Manager	88,–
Sybex ST Kontor Kundenverwaltung	135,–
Sybex ST Kontor Lager + Fakturierung	369,–
PRODATA fibuMAN e .. **375,–**	fibuMAN f .. **729,–**
GFA UTILITY'S	je 55,–
Gehäuse für NEC 1037 A einzel 24,–	doppel 34,–
ATARI Scartkabel ... 27,–	Druckerkabel ... 22,–
Floppykabel ATARI ST an 3,5" Laufwerk	27,–

Preisliste anfordern. Änderungen vorbehalten.
NEC Drucker und Monitore 12 Mon. Garantie. Deutsches Handbuch.

4600 Dortmund 15 · Kranenbusch 28
☎ 02 31/35 65 11

A COPY ST

★ Das Kopierprogramm ★

- Ein unentbehrliches Hilfsmittel zur erlaubten Sicherung Ihrer Originalsoftware. Raubkopien sind strafbar!
- Kopiert nach neuesten Verfahren.
- V1.2L sehr leistungsfähiges Kopierprogramm, reine Softwarelösung und dadurch kein umständlicher Umbau Ihres Systems.
- Macht dem Diskcontroller WD 1772 Beine.
- Voll GEM-unterstützt, dadurch sehr einfach in der Handhabung.
- Automatische Fehlererkennung, dadurch keine Parameterangabe notwendig.
- Erstellt bei normal Kopieren automatisch Schnelladedisketten.
- Eigene Formatierroutine gibt bis zu 230 KB bzw. 130 KB mehr Diskettenkapazität (Schnellade-Format).
- Multiple Option (Mehrfach-Kopien ohne neu einlesen).
- Abschaltbare Verify-Option.
- Update Service bis V1.3 ohne Kosten (nur Porto).
- Mit ausführlicher Anleitung.
- Achtung! A Copy hält, was es verspricht!!!!

nur DM 69,–

EUROSYSTEMS

FILIALE FÜR DEUTSCHLAND:
BAUSTRASSE 4, 4240 EMMERICH,
TEL. TÄGLICH 15-17.30 UHR 0 28 22/4 55 89

BESTELLUNG: BEI VORKASSE. VERSANDKOSTEN 6.– DM
PER NACHNAHME: VERSANDKOSTEN 8.– DM
UNABHÄNGIG VON DER BESTELLTEN STÜCKZAHL.
AUSLAND: NUR VORKASSE, EUROCH., POSTANW.
LIEFERUNG: 48-STUNDEN-SERVICE (WENN LAGERND)
BESTELLUNG RUND UM DIE UHR MÖGLICH

PRINT TECHNIK

8000 München 40 · Nikolaistr. 2 · Tel. 0 89/36 81 97 · FAX: 0 89/39 97 70

PRINT-TECHNIK UNIVERSAL ST-SCANNER DM 1498,-

Fakten: **Die Preissensation von Düsseldorf**
Der Scanner kann gleichzeitig als Bilderfassungsgerät, Kopierer und Drucker eingesetzt werden. Die Druckdichte ist 8 Punkte/mm oder 200 Punkte/Zoll. Läuft in allen ATARI ST-Bildschirmauflösungen. Im Scannbetrieb können DIN-A4-Vorlagen mit einer Auflösung von 200 Punkten pro Zoll erfaßt werden. Die Bilderfassung dauert nur 10 Sek. Ausschnittvergrößerungen und Speichern der Bilder in nahezu jedem Format ist selbstverständlich. Über GDOS-Treiber ist Thermodruck über alle entsprechenden Programme, wie Timeworks, GEM-Paint, GEM-Draw usw. möglich.
Der Scanner wird mit Software geliefert. **2998,-**

Weiterhin enthält die Software „ROGER PAINT" ein komplettes Zeichenprogramm, daß keine Wünsche offen läßt. Auch Druckertreiber für Atari Laser, P6 & P7 etc. vorhanden. Einbindungsroutine für CALAMUS jetzt in der Software enthalten.

NEU!!!
PRINT TECHNIK PROFESSIONAL SCANNER
300 dpi NUR Scanner mit extrem hoher Auflösung und direct dpi transfer für 300 dpi Laserdrucker. Software entspricht sonst der des Universal Scanners. Reine Profianwendung incl. OCR-JUNIOR

OCR SOFTWARE	**DM 698,-**
ROMPORTSTECKER freier Druckerport ermöglicht Sofortausdruck	**DM 198,-**
VIDEO DIGITIZER PRO 8805 Auflösung bis zu 1024×512 + 128 grau. Langsamer hochauflösender Digitizer für professionelle Anwendung	**DM 498,-**
VIDEO DIGITIZER REALTIZER Schneller Digitizer für 320×200 und 640×400 unterstützend	**DM 198,-**

(Beide Digitizer unterstützen alle gängigen Zeichenformate und Desktop Publisher und verfügen über ein Tool zum Verändern des Bildes. Kompatibel mit s/w und Colorkamera sowie VCR.)

VIDEO-TEXT-EMPFANGS-MODUL Dieses Modul erlaubt in Verbindung mit der Software den VIDEO TEXT Ihres Fernsehers oder VCR's auf dem Bildschirm des ATARI darzustellen, auszudrucken und abzuspeichern. Empfängt alle Programme, auch Sky Channel und Kabelprogramme	**DM 298,-**
ATARI GENLOCK	**DM 1498,-**
ATARI GENLOCK Studioversion Auf Studiomaschinen getestet. Das professionelle Genlock für den Atari User.	**DM 3498,-**
SPEICHERSCOPE	**DM 898,-**
KOMPL. METEO-SAT EMPFANGSANLAGE	**DM 3498,-**

Demodisk: DM 15,- ● Katalog anfordern! (DM 3,-) ● Täglich Versand

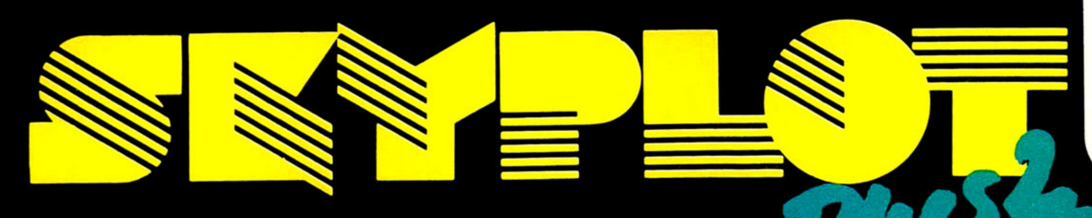
ÖSTERREICH · 1060 WIEN · STUMPERGASSE 34 · TEL. 0222/597 34 23 · TELEX 112 996

SCHWEIZ · MICROTRON · 2542 PIETERLEIN · BAHNHOFSTR. 2 · TEL. 032/87 24 29

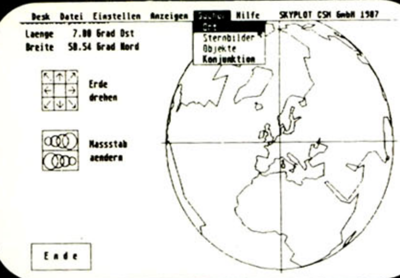
Benelux: 010-450 76 96 / NL: 079-41 25 63

STARKE SOFTWARE

DER GESTEIGERTE WAHNSINN:



Plus 2



Bestimmen des Standortes

Desk Datei Einstellen Anzeigen Suchen Hilfe SKYPLOT CSH GmbH 1987
Art der Darstellung
Datum und Uhrzeit
Ort
Sterne
✓ Planeten
Nebel
✓ Offene Sternhaufen
✓ Kugelsternhaufen
✓ Planetarische Nebel
✓ Diffuse Nebel
✓ Galaxien
Gradnetz
Grenzgröße
Ellipt. Bahnelemente
Parabol. Bahnelemente
Zeigerposition
Farben

Was wird dargestellt (hier mit 15.384 Sternen)

TECHNISCHE DATEN IN KÜRZE:

- ★ läuft auf ATARI ST mit mind. 1 MByte, Monochrom- oder Farbmonitor
- ★ zwei Sterndatensätze mit 613 bzw. **15.384 Sternen**
- ★ **1054 Nebel, Sternhaufen** und **Galaxien**
- ★ Daten im ASCII-Format, erweiter- und veränderbar
- ★ **8 Planeten, Sonne, Mond** und 3 Kometen fest eingestellt
- ★ zusätzlich Bahnelemente von **17 Kleinplaneten** und **14 Kometen**
- ★ Eingabe **eigener Bahnelemente** möglich, **elliptisch** und **parabolisch**
- ★ Speichern und Laden aller eingestellten Werte, auch der Bahnelemente
- ★ **6 Kartenarten:** sichtbarer Himmel, Horizont-, Polar-, Äquatorial-, Übersichts- und Umgebungskarte
- ★ **Vergrößerung** auf rechteckigen Karten mit der Maus
- ★ **maßstäbliche Darstellung** der Planeten **mit Phasen**
- ★ Datumsbereich **4713 v. Chr.** bis **22666 n. Chr.**
- ★ Präzissions- und Parallaxenkorrektur
- ★ geographische Breite und Länge einstellbar
- ★ Zeiteingabe als Weltzeit, Ortszeit oder Julianisches Datum
- ★ **3-D Darstellung** und Himmel von Planeten anderer Sterne
- ★ **viele Diagramme:** Finsternisverlauf, Hertzsprung-Russel-Diagramm, Differenzbewegung zur Sonne, Sichtbarkeit
- ★ **Suchen** nach **Konjunktionen, Oppositionen, größten Elongationen, Sternbedeckungen** durch den Mond, **Finsternissen...**
- ★ Planetariumsprogramm eingebaut und einzeln
- ★ Speichern und Laden von Bildern im komprimierten Format, DOODLE oder DEGAS
- ★ komprimierte Bildsequenzen laufen auch ohne Hard- oder Ramdisk schnell ab und brauchen wenig Speicherplatz
- ★ Objektgruppen einzeln abschaltbar
- ★ Sterne nach Spektralklassen selektierbar
- ★ **Gradnetz** mit einstellbarem Abstand
- ★ Berechnung von Kulmination, Auf- und Untergang, Dämmerungszeiten
- ★ **Stellarstatistik** zählt Sterne auf Ausschnitt und Gesamthimmel
- ★ und vieles mehr...

Update
nur DM 50,-

SKYPLOT *Plus 2*
nur DM 198,-



Finsternisverlauf

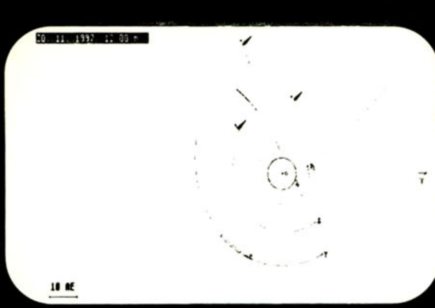
- Handbuch ca. **290 Seiten,** besonders auch für den Anfänger geschrieben
- wesentlich genauer und trotzdem **schneller** als SKYPLOT PLUS

Ansicht des Sonnensystems aus verschiedenen Blickwinkeln **Maßstab** und **Winkel** beliebig einstellbar

- **Planetenlauf** zeigt bewegte Planeten, heliozentrisch oder genozentrisch mit einstellbarem Winkel und Maßstab
- Starten anderer Programme, ohne SKYPLOT verlassen zu müssen
- **verschiedene Hardcopy-Formate,** auch gedreht
- **Suchen** nach Objekten und Sternbildern mit Wildcards
- ...und vieles mehr, wofür hier kein Platz mehr ist



Invertierter Himmel mit 613 Sternen + Planeten



Sonnensystem in Bewegungssimulation

Preise sind unverbindlich empfohlene Verkaufspreise.

SO ETWAS GAB'S FÜR DEN ATARI NOCH NIE!

BESTELLCOUPON

einsenden an
Heim Verlag · Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir ____ Stück **SKYPLOT PLUS 2** für nur **DM 198,-**\ St. ____ Stück **UPDATE** für nur **DM 50,-**\ St.
zzgl. DM 5,- Versandkosten (unabhängig von der Bestellmenge). Zahlung: ☐ Nachnahme ☐ Scheck liegt bei ☐ per Vorauszahlung

Name ____________ Vorname ____________ PLZ, Ort ____________

Str., Hausn. ____________ Unterschrift ____________

Benutzen Sie auch die in ST-COMPUTER vorhandene Bestellkarte.

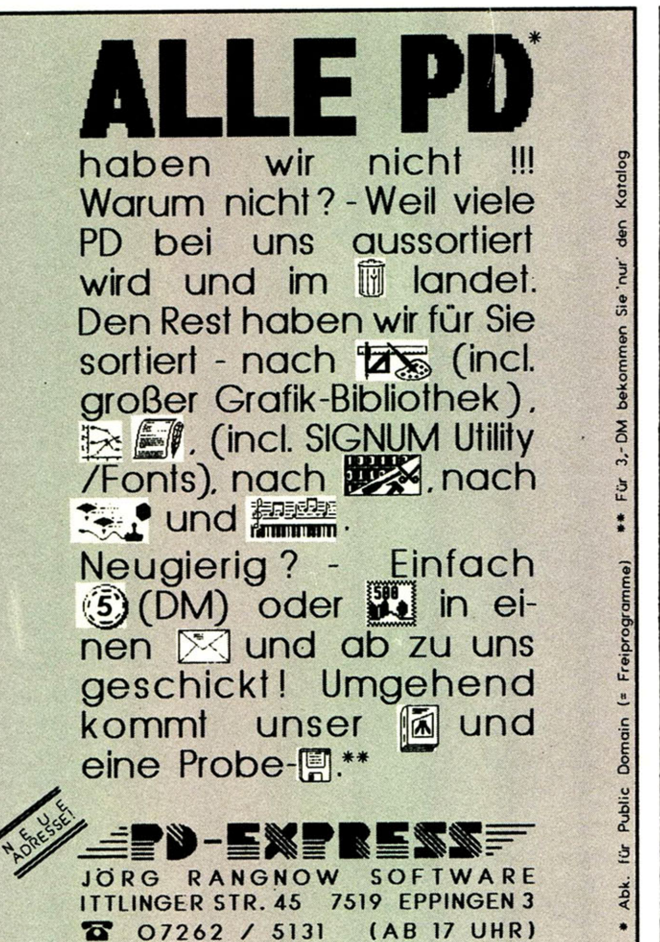
Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

Immer up to date

Mit dieser Sparte wollen wir allen unseren Lesern die Möglichkeit geben, sich über die neuesten Programm-Versionen zu informieren. Angegeben werden die aktuelle Versionsnummer, ein eventueller Kopierschutz, die Bildschirmauflösungen und der Speicherbedarf. Softwarefirmen ist es somit möglich, die ST-Computer-Leser über ihre Updates zu informieren.

Programmname	Version	Daten	Programmname	Version	Daten
Adress ST / Check ST	1.0	N H	K-Resource	1.1	N HM
AnsiTerm	1.4	N	Label ST	1.0	N HML
1st BASIC Tool	1.1	N HML	Laser C (Megamax)	1.2	N HML
Binlook	1.0	N HML	Laser Deluxe	1.0	N HML 1M
BTX für ST	1.0	N H 1M	1st_Lektor	1.2	HM
BTX/VTX-Manager	3.0	N H 1M	Lern ST	1.22	N HML
Calamus	1.01	N H 1M	Link_it GFA	1.1	N HML
CIS-L&G	1.01		Link_it Omikron	2.0	N HML
Crypt_it	1.0	J HML	Lisp Complete	1.01	N HM
fibuMAN	3.0	N H	Lock_it	1.0	J HML
fibuSTAT	2.3	N H	Mega Paint	1.0	N H 1M
Flash-Cache/Flash-Bak	1.0	N HM	Megamax Modula 2	3.5	N HM
Flexdisk	1.2	N HML	Micro C-Shell	2.70	N HM
1st_Freezer	1.0	N HML 1M	MT C-Shell	1.20	N HM 1M
GFA-Artist	1.0	N L	Multi ST	1.0	N HML 1M
GFA-Assembler	1.2	N HML	Musix32	1.01	J H
GFA-BASIC 68881	1.3	N HML	Omikron BASIC Compiler	3.01	N HML
GFA-BASIC-Compiler	2.02	N HML	Omikron BASIC Interpreter	3.0	N HML
GFA-BASIC-Interpreter	3.04	N HML	PAM's TERM/4014	3.012	N H
GFA-Draft	2.1	N	PAM's TurboDisk	1.7	N HML
GFA-Draft plus	3.0	N	PAM's NET	1.0	N HML
GFA-Farb-Konverter	1.2	N H	PC ditto Euroversion	3.64	N HML
GFA-Monochrom-Konverter	1.2	N ML	Pro Sound Designer	1.2	L
GFA-Objekt	1.2	N HM	Pro Sprite Designer	1.0	L
GFA-Raytrace	1.5		Search!	2.0	N HM
GFA-Starter	1.1	N HML	Signum! zwei	1.0	N H
GFA-Vektor	1.0	N	1st_Speeder	1.01	N HML
Hard Disk Accelerator	1.0	N HML	1st_Speeder 2	1.0	N HML 1M
Hard Disk Toolkit	1.05	N HM	STAD	1.3	N H
Harddisk Utility	2.0	N HM	ST Pascal plus	2.02	N HM
Imagic	1.1	N HML	Tempus	2.0	N HM
Intelligent Spooler	1.01	N HML	Transfile ST plus	3.0	N HM
Interlink	1.85	N HM	VSH Manager	1.11	N HML 1M

Irrtum vorbehalten!
Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung, L = niedrige Auflösung, 1M = mindestens 1 Megabyte

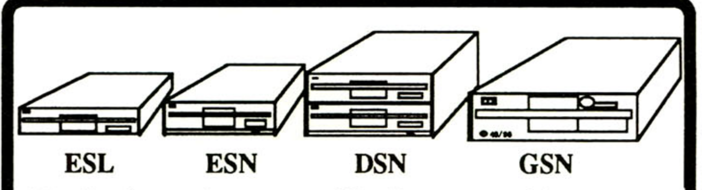






Unverbindliche Preisempfehlung:
Cashflow DM 298,–;
Handbuch DM 30,–*;
Demo 10,–;
*wird bei Direktkauf angerechnet.

C.A.$.H. GmbH
Robert-Bosch-Straße 20 a
8900 Augsburg
Telefon 08 21 / 70 38 56



ESL ESN DSN GSN

Alle Stationen sind anschlußfertig, doppelseitig, garantiert kompatibel, auf 83 Spuren/10 Sektoren geprüft, haben ein formschönes, hochwertiges Metallgehäuse und die einzigartige automatische Netzanschaltung.

ESL:	3,5"- Einzelstation mit FD1037A, Steckernetzteil, 28 * 105 * 165 mm	**269,-- DM**

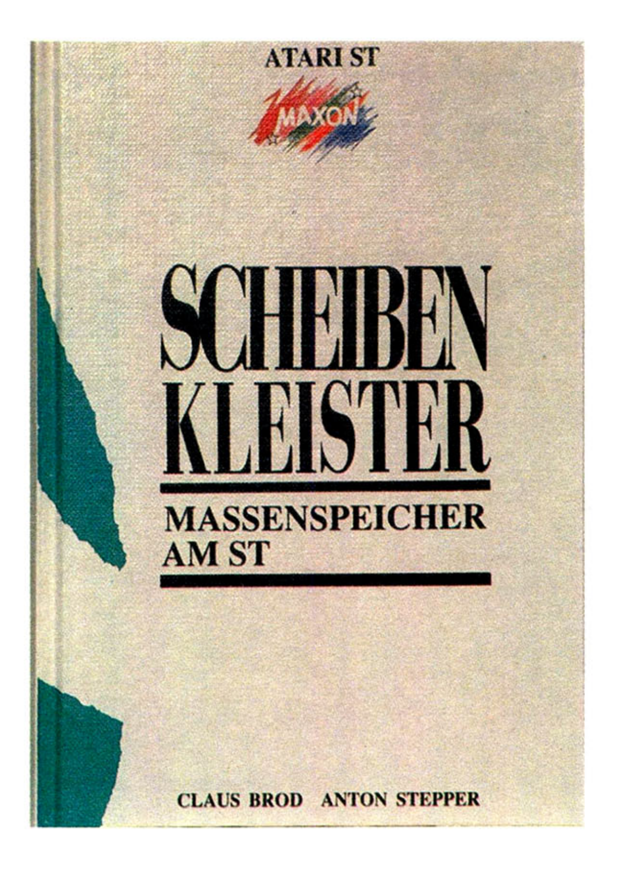
Auf die Luxusklasse mit dem bewährten NEC FD 1036A/ TEAC FD55FR, Netzkontrollanzeige, eingebautem Netzteil **geben wir 12 Monate Garantie:**

ESN:	3,5"- Einzelstation, 42 * 108 * 230	**299,-- DM**
ESN/A:	dto mit Ausgang für Laufwerk B	**339,-- DM**
ESN/AB:	dto mit automatischer Umschaltung für 2. B - Laufwerk, also 3 LWs. am ST	**379,-- DM**
DSN:	3,5"- Doppelstation, 75 * 106 * 230	**498,-- DM**
DSN/B:	dto mit Ausgang für 2. B - Laufwerk, wie ESN/AB	**578,-- DM**
GSN/2:	**5,25"** - Einzelstation, umschaltbar auf 40 Spuren, 50 * 152 * 290 mm	**369,-- DM**
GSN/3:	dto mit Ausgang für 3 Laufwerke am ST, paßt in jedem Fall	**429,-- DM**

Dipl.Ing. Gerhard Trumpp
Mitterlängstrasse 7
8039 Puchheim - Ort

Tel. 089 / 80 68 23
von 17 - 22 Uhr

Lesen Sie Scheibenkleister!



SCHEIBENKLEISTER MASSENSPEICHER AM ST

Alles über Floppies, Festplatten und andere Massenspeicher am ST von Claus Brod und Anton Stepper.

Was steht drin?

Kursteil (für die ganze Familie):
- Floppyprogrammierung mit allen erlaubten und unerlaubten Mitteln (per BIOS, XBIOS, GEMDOS und direkter Controllerprogrammierung)
- Kopierschutz, Aufzeichnungsverfahren, Datenstrukturen auf der Diskette
- Hardwaredokumentation zu Floppy und Festplatte (Anschluß von Fremdlaufwerken, Justierung, Reparaturhinweise)
- Festplatte: Prinzip, Controller, Programmierung

Nachschlageteil (für Programmierer):
- Hard- und Softwarereferenz zu DMA-Chip, Floppycontroller, Festplattencontroller
- **GEMDOS**-, **BIOS**- und **XBIOS**- Funktionen zur Massenspeicherprogrammierung (auch als GFA-BASIC-Bibliothek auf Diskette)
- Systemvariablen (auch bisher undokumentierte), physikalische Grundlagen, Pinbelegungen und Ports.

Software (für alle, fertig zum Anwenden mit Anleitungen):

- **TED**, der Trackeditor: Formate analysieren, ändern, erstellen; Zugriff auf alle Controllerfunktionen

- **SED**, der Datei- und Sektormonitor für **RAM-Disk**, **EPROM-Disk**, Floppies und Festplatten: Ordnernamen ändern, gelöschte Dateien retten, spezieller Harddiskmonitor für direkten Festplattenzugriff (eigene Formatierroutine für max. drei MB mehr)
- neue HYPERFORMAT-Version 3.0: Bis zu 950 KB auf doppelseitiger Diskette, superfixe Formatierroutinen (optional unter 20 Sekunden für doppelseitige Disketten), Schnelladeformate
- Steprateneinstellung, Konvertierung von Disketten auf Schnelladeformat
- Assemblerroutinen für direkten Floppy- und Festplattenzugriff zum Einbinden in eigene Programme
- lauffähige Programme mit Quelltext auf Diskette.

582 Seiten, Buch mit Diskette für DM 59.-



Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: ______________________

Vorname: ______________________

Straße: ______________________

Ort: ______________________

Unterschrift: ______________________

Hiermit bestelle ich:

☐ Exemplare von "Scheibenkleister, Massenspeicher am ST" mit Diskette für DM 59,00

Versandkosten: DM 7,50
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr.

☐ Vorauskasse
☐ Nachnahme

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3

ST-COMPUTER PUBLIC DOMAIN

Es weihnachtet sehr,

haben Sie schon einmal überlegt, was Sie Ihrem Computer zu Weihnachten schenken? Wenn Sie sich mit Leckereien den Bauch vollschlagen, was bekommt dann der graue Freund? Aber Ernst beiseite, in diesem schon garstig kalten Wintermonat, gibt es in unserer PD-Sammlung wieder einige Programme, die Ihnen und Ihrem Rechner kräftig einheizen. Werfen Sie doch einfach einmal ein Blick auf die folgenden Seiten oder ziehen Sie sich die zweite Ausgabe der PD-NEWS zu Gemüte.

Ihr MAXON PD-Service

Neuheiten



DRUCKUTILITIES



BIG BANNER: Programm zum Drucken von Endlos-Bannern. Erzeugt meterlange Schriftzüge in verschiedenen Schriftarten und -größen. (s/w)

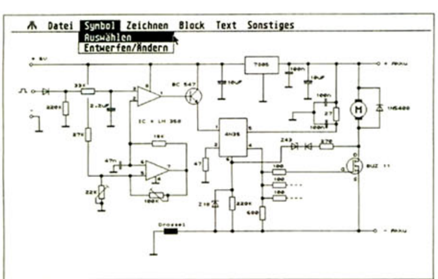
LC10 COLOR: 1st_Word-Druckertreiber für STAR LC10 Color. Ermöglicht vielfarbigen Textausdruck. Auch für andere Farbdrucker anpassbar.

P2000_INST: Einstellprogramm für sämtliche Steuercodes per Accessory. Zeilenabstand, Schrifttyp, Formatierung, Seitengröße,...

NL10_INST: Weiteres Einstellprogramm für NL10. Als Accessory kann der Drucker jederzeit nach eigenen Wünschen konfiguriert werden. (s/w)

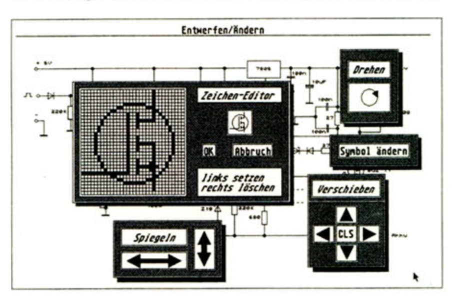


ELEKTRONIK



Motorsteuerung mit TRANSISTOR gezeichnet.

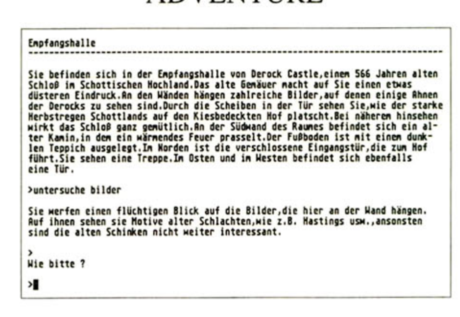
TRANSISTOR: Entwicklung von elektronischen Schaltungen. Ausgeklügeltes Zeichenprogramm mit umfangreicher Symbolbibliothek. Integrierter Symboleditor (Spiegeln, Verschieben, Drehen). Dadurch lassen sich auch ausgefallene Bauteile generieren. Blockoperationen, Lupe und sonstige Standardfunktionen. Die abgebildete Schaltung läßt sich in kurzer Zeit anfertigen. (s/w)



Ein neues Bauteil läßt sich leicht erzeugen.



ADVENTURE



Empfangshalle

Sie befinden sich in der Empfangshalle von Derock Castle,einem 566 Jahren alten Schloß im Schottischen Hochland.Das alte Gemäuer macht auf Sie einen etwas düsteren Eindruck.An den Wänden hängen zahlreiche Bilder,auf denen einige Ahnen der Derocks zu sehen sind.Durch die Scheiben in der Tür sehen Sie,wie der starke Herbstregen Schottlands auf den Kiesbedeckten Hof platscht.Bei näherem hinsehen wirkt das Schloß ganz gemütlich.An der Südwand des Raumes befindet sich ein alter Kamin,in dem ein wärmendes Feuer prasselt.Der Fußboden ist mit einem dunklen Teppich ausgelegt.Im Norden ist die verschlossene Eingangstür,die zum Hof führt.Sie sehen eine Treppe.Im Osten und im Westen befindet sich ebenfalls eine Tür.

>untersuche bilder

Sie werfen einen flüchtigen Blick auf die Bilder,die hier an der Wand hängen. Auf ihnen sehen sie Motive alter Schlachten,wie z.B. Hastings usw.,ansonsten sind die alten Schinken nicht weiter interessant.

>
Wie bitte ?

>

THE VAULT: In einem alten schottischen Schloß wird eines Morgens der Schloßherr an einem Baum erhängt aufgefunden. Die Indizien deuten auf einen Mord, doch die Polizei tappt im Dunkeln. Einige Hinweise sind gegeben, die restlichen müssen Sie im Schloß finden.

Es empfiehlt sich das Kartographieren der vielen Gänge, denn sonst verirrt man sich und wird den Fall nie lösen.

Das Adventure ist komplett in deutscher Sprache geschrieben und soll Ihnen helfen, die Winternächte zu überstehen.

Die Redaktion nimmt Lösungen des Spiels gerne entgegen.



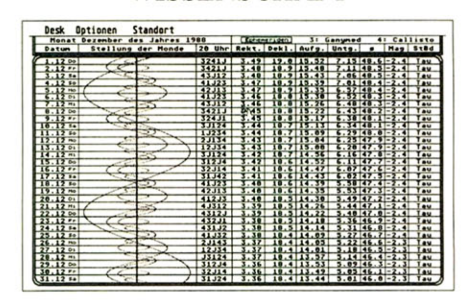
SPIEL



EUROPA: Actionspiel mit Joysticksteuerung. Eine finstere Gestalt hat in Europa mehrere Bomben verteilt, die Sie als tapferer Agent finden, entschärfen und vernichten müssen. Dazu sind vier Level zu bewältigen, bei denen viel Fingerspitzengefühl gefragt ist. (s/w)



WISSENSCHAFT



EPHEMERIDEN: Das Programm dient dazu, nach Eingabe eines bestimmten Zeitraumes innerhalb eines Jahres die Ephemeriden verschiedener Objekte zu berechnen. Zusätzlich wird die Möglichkeit geboten, die Mond- und Sonnenfinsternisse eines Jahres zu berechnen, sowie die Konstellation der vier inneren Jupitermonde auf den Bildschirm zu plotten.

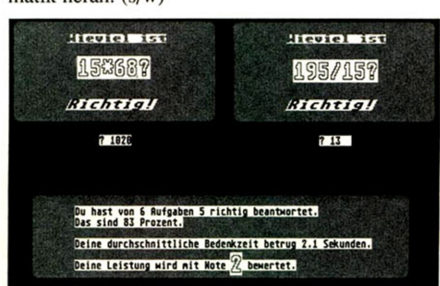
Programm mit langer Sternwartenerfahrung. (s/w)



SCHULE



ARTUS: Rechentrainer für die unteren Schulklassen. ARTUS, der Rechentrainer, führt mit netten Auflockerungen an das leidliche Thema Mathematik heran. (s/w)



KOPFRECHNEN: Rechentrainer für die Rechenarten +,-,*,/ und Wurzel. Verschiedene

Schwierigkeitsstufen von leicht bis reichlich schwer. Als Gegenpol für Taschenrechner-verwöhnte Hirne. Nicht nur für Kinder. (s/w)



RECHEN: Wirtschaftsrechnen für Bürokaufleute. Umrechnen von Maßeinheiten, Mathematik, Dreisatz, Prozentrechnen, u.v.a.m.

FASTTEST: Reaktionstrainer. Einfaches Programm zum Überprüfen Ihres Reaktionsvermögens. (s/w)

STUNDENPLAN: Stundenplan-Designer für Schüler und Lehrer (mehrere Klassen). (s/w) (mit GFA 2.0-Source)



SPECKWEG: Der freundliche Helfer zur Beseitigung Überflüssiger Pfunde. Generiert Abnahmeplan und informiert über den Kaloriengehalt vieler Nahrungsmittel. Druck einer Kalorientabelle. (s/w)

ALKOHOLTEST: Informatives Programm über Alkoholwerte.
Unter Berücksichtigung persönlicher Körpermaße berechnet das Programm den theoretischen Blutalkoholwert, die Abnahmekurve und informiert über Wirkung und Gefahren. Weiterhin wird die Anfälligkeit durch vielzählige Fragen geurteilt. (s/w)

Achtung: Beide Programme ersetzten keine Fachberatung!!





Sonderdisk C

EXTENDED VT52-Emulator

Wer es bis jetzt noch nicht gemerkt hat, unser EXTENDED VT52-Emulator wird neuerdings als Sonderdisk C für DM 15,- angeboten. Nähere Informationen kann man auf der nächsten Seite finden.

Auf der nächsten Seite
finden Sie weitere Disketten
aus unserer Sammlung



Wir verwenden ausschließlich doppelseitige Markendisketten (2DD) der Firmen Maxell und JVC. Sämtliche Disketten sind geprüft, fehlerfrei und virenlos.

Sparen Sie Porto!!!

Die Original-Disketten unserer Public Domain-Sammlung können Sie auch bei folgenden Händlern direkt mit Erscheinen der jeweiligen ST Computer erhalten:

Computer Corner Preetz
Hohen Kamp 2
2308 Preetz

Intersoft
Nohlstr. 76
4200 Oberhausen

OCB
Wallstr. 3
4422 Ahaus

Weber Bürotechnik
Naturparkstr. 16
5940 Lennestadt

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt/M.

Jacom Computertechnik
Hertz Str. 1
6900 Heidelberg

Leonhardt Electronic
In der Jeuch 3
7600 Offenburg

Udo Meier
Ringstr. 4
7700 Singen

resin Computershop
Hauptstr. 192-4
7858 Weil am Rhein

SCHULZ Computer
Schilerstr. 22
8000 München 2

A&P Shop Steuer
Auf der Schanz
8490 Cham

Decker Microcomputer
Meisenweg 29
8520 Erlangen

EDV-Baumann
Jean-Paul-Str. 16
8580 Bayreuth



PD-NEWS 2 ist erhältlich bei:

MAXON Computer GmbH
'PD-Service'
Industriestr. 26
6236 Eschborn

In PD-NEWS 2 lesen Sie unter anderem folgende Artikel:

- **Die Hyperstory:** Wissenswertes über Hyperformat und dessen Autor Claus Brod
- **SAKROTAN:** Computerviren und deren Beseitigung
- **Hochzeit:** Ein Beispiel linearer Optimierung mit ORSPREAD
- **KI zum Nulltarif:** XLISP 2.0 und deren Folgen
- **FSELECT:** Geheimnisse des neuen Fileselect-Standards

Programmierpraxis: Hardcopy ohne Streifen (GFA-BASIC)

Hardwarebastelei: JOYSTICK mit Dauerfeuer - Schonung des Daumens

Unterhaltsames: Computer und die Nachbarn - ein Bericht aus den täglichen Leben eines Computerfreaks

(Änderungen vorbehalten!)

Auch als **Nachschlagewerk**:
Enthalten ist eine komplette, ausführliche Liste unserer PD-Sammlung. Hier finden Sie jedes Programm mit Kurzbeschreibung, von 1-175.

Preis: DM 2.50 (+ DM 1.50 Porto)
Bei Bestellung von PD-Disketten enfallen die DM 1.50 Porto. Ab 5 Disketten gibt es die PD-NEWS 2 gratis dazu.

ST-COMPUTER PUBLIC DOMAIN

Altbewährtes

Die nicht aufgeführten Disketten sind natürlich auch weiterhin erhältlich, nur wurde uns der Platz zu klein. Schauen sie dazu bitte in einer der vorherigen Ausgaben nach oder bestellen Sie die PD-NEWS.

Diskette 166
Fremdsprachen
-**ECS**: Lernprogramm für unregelmäßige englische Verben. Konjugiert Verben in die gewünschte Zeit. Viele Verben enthalten.
-**VOCMAN**: Umfangreiches Vokabelprogramm. Verwaltet bei Vokabeln mehrere mögliche Übersetzungen. Z.B.: car auto automobile Auto Automobil voiture coche
Flexible Organisation, Aufbau eigener Datenbanken
-**VOKABEL**: Lernprogramm bzw. Wörterbuch. Als Beispiel ist eine Botanik-Datei enthalten, die die lateinischen Bezeichnungen und die deutschen Namen enthält. Dies zeigt, daß damit nicht nur Fremdsprachen gelernt werden können. Falsche Antworten werden in eine Fehlerdatei übernommen, um das Gelernte zu überprüfen.

Diskette 167
Aufkleber
-**AUDIO 2000**: Aufkleber-Druckprogramm für Audio-Cassetten, LPs, CDs und DATs. Daten werden selbstverständlich zum späteren Gebrauch auf Disk abgelegt. (s/w)
FCP6_ETIKETT: Etikettenprogramm mit Farb-Druck. Wer einen Farbdrukker besitzt, kann nun seine Disketten farbig bedrucken. Jedes Element kann eine andere Farbe bekommen. Das wird im Menü neben den Druckdaten, den ICONS und den Diskinfos angeklickt.
Das Programm läuft natürlich auch mit normalen Druckern. (s/w)
-**MC_WRITER:** Druckprogramm für Musikkassetten. Neben der nützlichen Programmfunktion dient es auch als Demo für GEM-Programmierung im neuen OMIKRON.BASIC. (Source-Code enthalten)

Diskette 168
Simulationen
-**MANDELBROT**: Schnelle Berechnung der Mandelbrod'schen Menge durch besonderen Algorithmus. Druckausgabe mit erhöhter Auflösung, keine Hardcopy, sondern mit ca. 10facher Auflösung des Bildschirms. (s/w). Mit Source-Code in C.
-**LIFE**: Äußerst fixe Version der Fortpflanzungsberechnung. Zufallsmuster und Glider-Kanonen (residente Generationen). Bis zu 50 Generationen pro Sekunde.
-**GRAFCOMO**: Programm für grafisch begabte Mathematiker. Erzeugt hübsche Grafiken mit hochmathematischen Hintergrund. (s/w)
KALEIDOSKOP: Faszinierendes Spiel der Formen und Muster. (s/w)

Diskette 169
Spiele
-**LARN**: Adventure für die kalten Herbstabende. Dringen Sie, nachdem Sie sich eine Abenteurer-Ausrüstung organisiert haben, in eine unbekannte Höhle ein. Dort erwartet Sie eine Vielzahl von Gängen, Schätzen, Kobolde, Fallen, Zaubertränken und sonstigen Tücken. LARN ist sehr flexibel, wenn Sie wollen, können Sie sich z.B. das Labyrinth nach eigenen Wünschen kreieren. Hohe Spiel-Motivation.
-**ROCKET:** Rocket-Defense ist ein bekanntes Spiel, bei dem die Erde vor Angriffen aus dem Weltraum geschützt werden muß. Gespielt wird mit der Maus. Mit steigenden Level ist Konzentration gefragt. (s/w)

Diskette 170/171
IMAGIC BILDERWERKSTATT ATARI ST
-**IMAGIC:** Passend zu unserer Serie "Bildwerkstatt ATARI ST", die in Heft 11/88 der ST Computer begonnen hat, bieten wir eine Spezialversion des IMAGIC-Grafik-Compilers an. Natürlich ersetzt sie nicht das Originalprogramm, aber es ist alles möglich, bis auf den einen Punkt, eine fertige Show abzuspeichern. Somit kann man die komplette Serie leicht nachvollziehen, ohne sich gleich das Geld für das Originalprogramm auszugeben.
Sinnvoll ist es allerdings nur, beide Disketten (PD 170 & 171) zusammen zu bestellen, da auf der einen der Grafik-Compiler und auf der anderen die fertig digitalisierten Bilder und die nötigen Strukturen der Serie vorhanden sind. Beginnen auch Sie ein neues Zeitalter mit DTV (Desktop Video)!

> **Machen Sie mit!**
>
> Möchten Sie ein selbstgeschriebenes Programm in die PD-Sammlung geben? Kein Problem. Schikken Sie es uns auf einer Diskette zu, samt einer Bestätigung, daß es von Ihnen erstellt wurde und frei von Rechten Dritter ist.

Diskette 172
-OMIKRON UPDATE 3.0 -> 3.00
Allen glücklichen Besitzern des neuen OMIKRON.BASICs bieten wir im Zusammenarbeit mit der Firma OMIKRON. einen einmaligen Update-Service. Das Programm auf dieser Diskette dated Ihren OMIKRON.Interpreter V3.0 auf die neueste Version 3.00 up.
Neben kleinen Änderungen am Interpreter, generiert dieses Programm eine überarbeitete GEMLIB (belegt nur noch eine Programmzeile) mit unverzichtbaren Funktionen, und alle anderen Dateien, die zur Original-OMIKRON-Diskette gehören (siehe NEWS).
Achtung: Das Update-Progamm prüft nach, ob Sie version 3.0 schon besitzen und generiert nur dann die neue Version 3.0

Diskette 173
3D UND SIMULATIONEN
-**ERDKUGEL:** Realtime-Animations-Programm. Die Erdkugel wird mit 4 Bildern pro Sekunde im Raum bewegt. Dabei können Drehung, Lage und Entfernung per Tastatur manipuliert werden. "Scotty beamen".
-**LIFE 3D:** Eine räumliche Variante des Generationenspiels. Durch die dritte Generation ergeben sich neue Aspekte. (s/w)
-**EDIT 3D:** Programm zur räumlichen Darstellung von Körpern. Diese können frei definiert und im Raum bewegt werden. (s/w)
-**PENDEL:** Simulation eines Drehpendels. Der Bewegungsablauf wird mit Hilfe von DGLs berechnet und grafisch dargestellt. Interessantes Programm für's Physik-Labor zu Hause. (s/w)
-**FILTER:** Programm zur Berechnung verschiedenster Filter: IIR- und FIR-Filtertypen, Bessel-, Butterworth- und Tschebyschefffilter mit verschiedenen Welligkeiten. Dabei werden Band- Hoch-, Tief- und Allpaß berechnet. Neben den Werten wird der Signalpegel grafisch dargestellt. (s/w)

Diskette 174
SPIELE
-**TUZZLE:** Originelles Computerspiel. Durch Verschieben von Bausteinen (Rohrleitungen) muß dem einfließenden Wasser ermöglicht werden, von links nach rechts zu fließen. Dabei ist Überblick und schnelle Reaktionszeit gefragt, denn sonst ist das Wasser schneller. Ein unterhaltsames Spiel für Joystick und s/w-Monitor. (s/w)
-**PUSH_ME:** Spiel für scharfe Denker. Auf einem Spielfeld muß ein Viereck von der Start- auf die Zielposition geschoben werden. Das hört sich aber leichter an, als es ist. In der Redaktion waren jedenfalls einige Personen, viele Stunden lang, anstatt ihrer Arbeit nachzugehen, am Steine-Schieben.
(s/w)
-**BUMERANG:** Arkanoid-Variante mit Bumerang statt Ball. Dadurch ergibt sich ein besonderer Bewegungseffekt, denn durch Anschneiden des Bumerangs fliegt dieser eine Kurve. (s/w)
-**LANDER:** Notlandung eines Raumschiffs auf einem Planeten. Der Blick aus dem Cockpit zeigt die Entfernung und den Landezielpunkt. Durch geschickte Richtungs- und Schub-Steuerung müssen Sie nun versuchen das Raumschiff sicher zu landen. Wenn es nicht funktionieren sollte (passiert recht oft), gibt das Programm eine Auswertung des Absturzes. (s/w)

Diskette 175
ANTIVIRUS & UTILITIES
-**SAKROTAN 4.03:** VIREN auf Ihrem ST? Keine Panik. SAKROTAN erkennt sie und desinfiziert die betroffenen Disketten. SAKROTAN unterscheidet durch seine Bibliothek zwischen bösartigen VIREN und anderen Bootprogrammen (TOS-Lader, Aladin-Booter, 60Hz-Umschalter, u.a.). Weiterhin erkennt es LINK-VIREN, indem es das betreffende Programm auf kritische Sprünge analysiert.
Ein spezielles Boot-Programm schützt die Disketten vor erneuten VIREN-Befall. Kurz ausgedrückt: SAKROTAN schützt vor allen bislang bekannten VIREN. (s/w)
Ausführlicher Bericht zu Sakrotan in der neuen Ausgabe der PD-NEWS. Sofort mitbestellen - es lohnt sich!
-**BCS:** Das BASIC Construction Set unterstützt die Programmierung von eigenen Desktops mit GFA-BASIC. Durch ein eingebautes Malprogramm können die Menüs (Buttons und Verzierungen) auf dem Bildschirm gezeichnet und Icons plaziert werden. BCS erzeugt dann ein ablauffähiges GFA-BASIC-Programm, das diesen Desktop aufbaut und die eingebauten Buttons abfragt. (s/w)
-**G_C SHELL:** Shell für GFA-BASIC. Mit diesem Programm können Interpreter und Compiler bequem aufgerufen werden. (s/w)

Diskette 161
Aus rechtlichen Gründen mußten wir Diskette 161 ändern.
Das Spiel Hase & Igel ist ab sofort nicht mehr enthalten, dafür haben wir ein weiteres Autorennspiel aufgenommen:

AUTORENNEN
-**DAKAR:** Autorennspiel mit Rennstreckeneditor. Zwei Personen heizen auf der selbst erstellten Strecke mit verschiedenen Rennwagen um den Sieg. Dabei gilt es aber Engpässe und Hyper-Magneten auszuweichen. Streckenteile und Rennwagen veränderbar. (s/w)
-**DODG_EM:** Zwei Rennwagen auf Kollisionskurs. Nur durch schnelle Reaktion kann man ausweichen. Mit gesampelten Sound. Einfache Grafik, aber interessante Spielidee. (s/w)

Zeichenerklärung:
s/w = nur monochrom
f = nur Farbe
keine Angaben = s/w und f

Sonderdisks

Die folgenden Programme sind nicht Public-Domain.
Sie können aber bei uns bezogen werden.

A.) **TOS**: Das Betriebssystem auf Diskette vom 6.2.1986. Was tun, wenn Programme mit Blitter-TOS nicht laufen? Einfach das alte TOS von Disk laden.
Unkostenbeitrag DM 15.-

B.) **RCS**: Das Resource-Construction Set aus dem ATARI-Entwicklungspakets. Unverzichtbar bei der GEM-Programmierung.
Unkostenbeitrag DM 15.

C.) **EXTENDED VT 52 Emulator**

Schneller als der Blitter
Erweiterte Version des in der ST-Computer ausführlich vorgestellten Emulators. Das Programm ersetzt den im TOS integrierten VT 52-Emulator vollkommen. Enthält neue Routinen zur Bildschirmausgabe, die wesentlich schneller sind, als die im TOS eingebauten. Daher erfolgt eine Beschleunigung der Textausgabe um den Faktor 3-5.
Programmierung
Neben der Beschleunigung enthält der Emulator zusätzliche, programmierbare ESC-Funktionen, wie z.B. das Softscrolling (1 Pixel) nach allen vier Seiten. Wurde es eingeschaltet, so scrollt eine Zeile oder ein Textblock ohne Rucken. Ferner können die Textattribute (fett, unterstrichen, hell und kursiv) ohne GEM umgeschaltet werden. Auch die Übertragung von Grafik via VT 52 ist möglich.
Der Extended VT 52 Emulator wurde in vier Teilen ausführlich in der ST-Computer vorgestellt.
Diese Diskette ist nicht public-domain, kann aber über diesen Service von uns bezogen werden.
Unkostenbeitrag DM 15.-

BITTE BEACHTEN

Sämtliche Disketten können ab dem Erstverkaufstag der ST-Computer direkt bei der MAXON-Computer bezogen werden. Wir haben für Sie den schnellstmöglichen Versandservice eingerichtet. Lieferung innerhalb einer Woche.
Bitte beachten Sie folgende Punkte:

1. Schriftliche Bestellung

- Der Unkostenbeitrag für eine Diskette beträgt DM 10,-
- Hinzu kommen Versandkosten von DM 5,- (Ausland DM 10.-)
- Bezahlung nur per Scheck oder Nachnahme (Im Ausland nur Vorrauskasse möglich)
- **Ab 5 Disketten entfallen die Versandkosten (DM 5.- bzw. DM 10.-)**
- Bei Nachnahme zuzüglich DM 3,70 Nachnahmegebühr
- Bitte fügen Sie keine anderen Bestellungen oder Anfragen bei.

Adresse:

MAXON-Computer GmbH
"PD ST-Computer"
Postfach 5969
D-6236 Eschborn

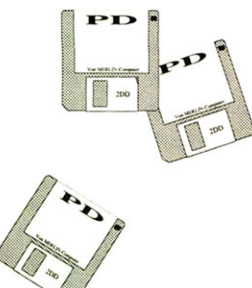
2. Anruf genügt

MAXON-Computer GmbH

"PD-Versand"
Tel.: 0 61 96 / 48 18 11
Mo-Fr 9⁰⁰ - 13⁰⁰ und 14⁰⁰ -17⁰⁰ Uhr

- Nur gegen Nachnahme (Gebühr DM 3,70)

Bei Fragen bezüglich der Programme stehen wir Ihnen gerne zur Verfügung.



Wann rufen Sie uns an?

Sie brauchen nur unsere Telefonnummer, denn wir denken, daß wir alles im Verkaufsprogramm haben, was es an Software für den ATARI ST gibt. Ehrlich!

Sehr verbraucherfreundlich: Wir geben auf alles Garantie!!!

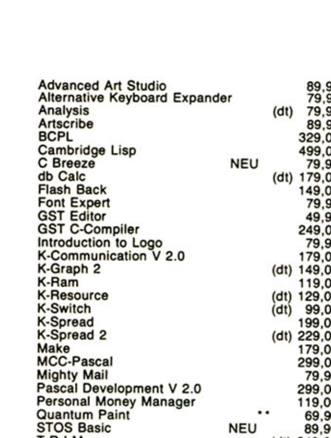
Wir führen: Alle Software (Auch die Neueste), alle Bücher, alle Anwendungssoftware, fast ohne Ausnahme.

Neu in der Top Ten, schon bei uns erhältlich.

Hier ein Auszug aus unserem Programm:

3 D-Galaxy		(dt)	59,90
19, Boot Camp		(dt)	64,95
1943	NEU		59,95
221 b Bakerstreet			79,90
500 cc Grand Prix			69,90
20.000 Meilen unter d. Meer		*(dt)	64,95
Action Service	NEU	(dt)	64,95
Alien Syndrome			59,95
Altair			59,90
Alternate Reality City		(dt)	69,90
Amazon			59,90
American Pool		(dt)	34,95
Arcade Force Four		(dt)	79,90
Arena			119,00
Arkanoid		(dt)	49,90
Arkanoid 2		(dt)	59,95
Artura	NEU	(dt)	64,95
Asterix im Morgenland		(dt)	64,95
Atax	NEU	(dt)	49,95
Autoduel			59,90
Backlash		(dt)	59,90
Bad Cat		(dt)	59,90
Balance of Power		*	99,00
Bard's Tale 1			89,00
Battleships		(dt)	49,95
Bermuda Project		(dt)	79,95
Better dead than Alien		(dt)	59,95
Beyond the Icepalace		(dt)	59,95
Beyond Zork		*	89,00
Bionic Commando		(dt)	59,95
Black Lamp			69,90
Blueberry: Das Gespenst...		(dt)	64,95
BMX-Simulator			49,90
Bob Morane Sience Fiction		(dt)	69,90
Bolo		*(dt)	69,00
Bolo Werkstatt	NEU	*(dt)	69,90
Bomb Jack			64,95
Bubble Bobble			59,90
Bubble Ghost		(dt)	59,95
Buggy Boy		(dt)	59,95
Captain America		(dt)	59,90
9Captain Blood		(dt)	74,95
Carrier Command		(dt)	84,95
Camonix Challenge		(dt)	69,90
Championship Baseball			89,90
Chese		*(dt)	89,90
Checkmate		*	39,90
Chopper X			29,95
Chronoquest	NEU	(dt)	99,90
Chubby Cristle		(dt)	64,95
Coruption		*(dt)	79,95
Clever und Smart		(dt)	59,95
Crazy Cars		(dt)	69,90
Cybernoids		(dt)	64,95
D.T. Olympic Challenge	NEU	(dt)	64,95
Dark Castle			89,00
Deathstrike		*(dt)	59,90
Defender of the Crown			79,90
Deja Vu			79,90
Dizzy Wizzard (1 MB RAM nötig)		*(dt)	59,90
Down at the Trolls	NEU	(dt)	59,95
Dr. Livingstone I Presume			49,90
Dungeon Master		(dt)	84,95
E C O (sphere)			59,90
Eden Blues		(dt)	89,00
Eddie Edwards Super Ski		(dt)	59,95
Electronic Pool		(dt)	69,90
Elemental	NEU	(dt)	59,95
Elite	NEU	(dt)	84,95
Empire Strikes Back			59,95
Enduro Racer			69,90
Epix Compilation			99,00
Euro Soccer 88		(dt)	59,95
Eye		(dt)	49,95
Fahrenheit 451		*	59,90
Fire Blaster		(dt)	39,90
Fire and Forget		(dt)	79,95
Football Fortunes			79,90
Fugger		(dt)	59,95
Fußball Manager 2		(dt)	64,95
Foundations Waste			79,95
Flight Simulator 2		*	179,00
Flight Sim. Scenery Japan			49,95
Flight Sim. Scenery Europa			49,95
Flight Sim. Scenery Disk 7			49,90
Flight Sim. Scenery Disk 11			49,90
Fred Feuerstein		(dt)	59,95
Freedom	NEU	(dt)	64,95
Gamblet		(dt)	49,90
Gato			99,00
Gary Lineker's Hotshots		(dt)	64,95
Gary Lineker's Superskills		(dt)	64,95
Gauntlet 2		(dt)	79,90
Gnome Ranger		*	49,90
Golden Path			69,90
Goldrunner 2		(dt)	59,90
Goldrunner 2-Scen. Disk 1			19,90
Goldrunner 2-Scen. Disk 2			19,90
Guild of Thieves		*	89,00
Hacker		*	69,95
Hardball			79,90
Harrier Strike Mission			89,00
Hellowoon		(dt)	69,95
Hollywood Hijinx		*	89,90
Hostages	NEU	(dt)	79,95
Hot Shot		(dt)	64,95
Impact			49,90
Impossible Mission 2			54,90
Intelligent Checkers		*(dt)	59,90
Indian Mission		*(dt)	59,90
Indoor Sports		(dt)	79,95
Inside Outing	NEU	(dt)	69,95
Jagd auf Roter Oktober		(dt)	84,95
Jet		*(dt)	119,00
Jinxter		*	79,95
Kampf um die Krone		(dt)	69,90
Kaiser		(dt)	129,00
Karting Grand Prix		(dt)	34,95
Killdozers		(dt)	59,95
Kings Quest 3er-Pack		*	84,95
Knight Orc		*	69,90
Leaderboard Birdie	NEU	(dt)	79,95
Leben und Sterben lassen	NEU	(dt)	64,95
Leisure Suit Larry			64,95
Legends of the Sword		*(dt)	79,95
Leviathan			49,90
Lombard RAC Ralley	NEU	(dt)	84,95
Luxor	NEU	(dt)	49,95
Lucky Luke: Nitroglyzerin		**(dt)	59,90
Mach 3			59,95
Marble Madness			89,90
Metropolis 2000 AD		(dt)	39,90
Mewilo		*(dt)	69,95
M G T		(dt)	89,00
Mickey Mouse		(dt)	59,95
Microleague Wrestling			69,90
Mindshadow		*	79,90
Moebius			79,90
Nebulus	NEU	(dt)	64,95
Netherworld		(dt)	64,95
Night Raider		(dt)	64,95
Nigel Mansell	NEU	(dt)	79,90
Nine Princess in Amber		*	59,90
Obliterator		(dt)	79,95
Off Shore Warrior	NEU	(dt)	59,95
Ogre			89,90
Oids		(dt)	59,90
Ooze		(dt)	79,95
Outcast		(dt)	39,90
Outrun		(dt)	59,90
Overlander	NEU	(dt)	64,95
Overlord	NEU		59,95
Pandora		(dt)	59,95
Pawn			89,90
Peter Pan	NEU	(dt)	64,95
Phantasm			59,95
Phantasie 3			79,90
Perry Mason		*	59,90
Pinball Factory			79,90
Pink Panther		(dt)	59,90
Plutos			59,90
Police Quest		*	64,95
Pool			34,95
Power Struggle		*	49,90
Q-Ball		(dt)	69,90
Quadralian		(dt)	59,90
Rana Rama			69,90
Return to Genesis		(dt)	59,90
Rings of Zilfin			79,90
Roadwars			69,90
Rockford			69,90
Rolling Thunder		(dt)	69,90
Rückkehr der Jedi Ritter	NEU	(dt)	64,95
Sargon Chess	NEU	(dt)	79,90
Seconds Out			69,90
Sentinel			69,95
Shackled		(dt)	59,95
Shuffleboard		(dt)	34,95
Shuttle 2			79,90
Side Walk		(dt)	69,90
Skyblaster		*(dt)	59,90
Sky Chess	NEU	(dt)	64,95
Skyfighter			49,90
Skyrider			69,90
Slapfight		(dt)	59,95
Slaygon Adventure		(dt)	69,90
Soldier of Light	NEU	(dt)	64,95
Sommer Olympiade		(dt)	64,95
Solomon's Key			59,90
Space Ace			69,90
Space Baller		(dt)	29,90
Space Harrier		(dt)	64,95
Space Port		(dt)	69,90
Space Quest 1		*	84,95
Space Quest 2		*	69,90
Space Station			79,90
Spitfire 40		(dt)	74,95
Spy versus Spy 1		(dt)	79,90
Starglider		*	89,90
Starglider 2		(dt)	84,95
Starray	NEU	(dt)	59,95
Startrash	NEU	(dt)	59,95
Startreck		(dt)	64,65
Starwars			59,90
Stellar Crusade			99,90
Streetfighter		(dt)	59,90
Strike Force Harrier			89,00
Strip Poker 2		**(dt)	49,90
ST Soccer		(dt)	69,90
ST Wars		(dt)	79,90
Sub Battle			79,90
Sundog	NEU		49,90
Super Hang On	NEU	(dt)	64,95
Supersprint			49,90
Superstar Icehockey		(dt)	89,00
Taipan		(dt)	59,90
Tanglewood		(dt)	59,90
Tau Ceti		(dt)	59,95
Tetraquest	NEU		59,95
Tee Up Golf			59,90
Terramex		(dt)	59,95
Terrorpods			79,90
Test Drive		(dt)	89,00
Tetris			69,90
Thai Boxing		(dt)	59,90
Thrust			34,95
Thundercats		(dt)	59,95
Tracker		*	89,90
Trailblazer		(dt)	79,90
Trantor		(dt)	59,95
Trashheap		(dt)	69,90
Trivial Persuit	NEU	(dt)	64,95
Turbo			29,90
Ultima 3			89,00
Ultima 4			89,00
Uninvited			89,00
Universal Military Simulator		*	84,95
Vampire's Empire		(dt)	59,95
Vegas Gambler		(dt)	79,00
Vermeer		*/**(dt)	79,90
Veteran	NEU	(dt)	49,95
Virus		(dt)	64,95
Wanted	NEU	(dt)	64,95
War Games Construction Kit			79,90
War Hawk			34,95
Warship	NEU		99,95
Warzone			29,90
Waterskiing		(dt)	59,95
Western Games		(dt)	59,95
Where Times stood still	NEU	(dt)	64,95
Whriligig	NEU	(dt)	64,95
Winter Games		(dt)	79,90
Winter Olympiad '88		(dt)	64,95
Wizball		(dt)	59,90
Xenon			69,90
3 D Video Studio		(dt)	148,00





Advanced Art Studio			89,90
Alternative Keyboard Expander			79,90
Analysis		(dt)	79,90
Artscribe			89,90
BCPL			329,00
Cambridge Lisp			499,00
C Breeze	NEU		79,90
db Calc		(dt)	179,00
Flash Back			149,00
Font Expert			79,90
GST Editor			49,90
GST C-Compiler			249,00
Introduction to Logo			79,90
K-Communication V 2.0			179,00
K-Graph 2		(dt)	149,00
K-Ram			119,00
K-Resource		(dt)	129,00
K-Switch		(dt)	99,00
K-Spread			199,00
K-Spread 2		(dt)	229,00
Make			179,00
MCC-Pascal			299,00
Mighty Mail			79,90
Pascal Development V 2.0			299,00
Personal Money Manager			119,00
Quantum Paint		**	69,90
STOS Basic	NEU		89,90
T R I M		(dt)	349,00

Telefon: 0 24 72 - 71 58

Odin-Software GmbH, Hengstbrüchelchen 39
D-5108 Monschau 4, Fax: 0 24 72 - 53 71

In der nächsten ST-Computer lesen Sie unter anderem

Lernen Sie Modula-2!

Modula-2 hat sich in der letzten Zeit als zweite Entwicklungssprache nach C auf dem ATARI ST etablieren können. Nicht nur die Flut der Compiler, sondern auch deren Qualität machen Modula-2 zu einer Sprache, die für ST-Benutzer immer größere Bedeutung erlangt. In den Ausgaben des vergangenen Jahres konnten Sie Tests der Systeme von TDI, Jefferson, Megamax, Softwave und SPC lesen - und alle zeigten sich als brauchbare Pakete.

ATARI und die Comdex Fall '88

"Comdex Fall" hieß die einhellige Antwort auf der Düsseldorfer ATARI-Messe. Gemeint war der Termin, an dem ATARI seinen neuen Rechner, den ATARI TT und sogar einen portablen ST präsentieren wollte. Ein paar Details waren bisher sogar schon zu erfahren. Um mehr zu recherchieren, haben wir keine Kosten und Mühe gescheut und haben uns kurzerhand auf die Reise begeben. Ob die angekündigten Produkte zu sehen waren und was es über sie zu berichten gibt, erfahren Sie in der nächsten Ausgabe der ST Computer.

Midi-Kommunikation - kurz MiKom

Jetzt, wo die Megas so richtig in die heimischen Computerstuben Einzug gehalten haben, finden sich sicherlich bei vielen ST-Freunden zwei STs. Aber nicht nur Besitzer zweier STs sollen angesprochen werden, sondern auch alle diejenigen, die mal zwei Rechner miteinander verbinden möchten. Wir stellen Ihnen ein Programm vor, das den MIDI-Ein- und Ausgang zur Kommunikation nutzt. Das Programm ist erweiterungsfähig, es lassen sich also gegebenenfalls noch viele Routinen implementieren.

Line A-Bibliothek für Turbo-C

Die meisten C-Compiler für den Atari ST werden mit umfangreichen Bibliotheken ausgeliefert, welche die Programmentwicklung wesentlich erleichtern. Jedoch fehlt bei den meisten Bibliotheken die Möglichkeit, die Line A-Funktionen aufzurufen. Für den Programmierer heißt das, will er beispielsweise einen Punkt auf dem Bildschirm setzen, eine eigene Routine zu programmieren oder sich dem langsamen und umständlichen VDI (Applikation anmelden, Workstation öffnen, VDI-Aufruf, Workstation schließen, Applikation abmelden) anzuvertrauen. Wir stellen Ihnen ein Programm vor, daß Abhilfe schafft. Es ermöglicht den Aufruf der Line A-Funktionen von Turbo-C aus. Zusätzlich läßt es sich mit geringem Aufwand auch an jeden anderen C-Compiler anpassen.

Änderungen vorbehalten!

Die ST Computer-Ausgabe 1'89 erscheint am 23.12.1989

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir ja weiterhin vorhaben, die Qualität zu steigern (ja, auch wenn das manchmal danebengeht), haben wir Redakteure ein großes Anliegen an Sie, liebe Leserinnen und Leser:

Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur Donnerstags von 14⁰⁰-17⁰⁰ Uhr telefonisch beantwortet werden können.

Vielen Dank für Ihr Verständnis

Impressum ST Computer

Chefredakteur:
Uwe Bärtels (UB)

Redaktion:
Uwe Bärtels (UB)
Harald Egel (HE)
Marcelo Merino (MM)
Harald Schneider (HS)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO)
Fernando Brand (FB)
Claus Brod (CB)
Stefan Höhn (SH)
Raymund Hofmann (RH)
Oliver Joppich (OJO)
Jürgen Leonhard (JL)
Claus P. Lippert (CPL)
Markus Nerding (MN)
Werner Schiewitz (WS)
Chr. Schormann (CS)
R.Tolksdorf (RT)

Autoren dieser Ausgabe:
A.Beller
V.Brixius
J.Drücker
R.Egelseer
A.Esser
M.Fangmeyer
M.Groneberg
M.Kraus
H.U.Mayer
F.Melchior
Dr.K.Sarnow
F.Schott
M.Wielebinski
C.D.Ziegler

Auslandskorrespondenz:
C.P.Lippert (Leitung)
D.dela Fuente (UK)
L.Hennelly (Nordamerika)

Redaktion: "MAXON" Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 11
FAX : 0 61 96/4 11 37

Verlag:
Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57
FAX : 0 61 51/5 56 89 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Margaritis

Anzeigenpreise:
nach Preisliste Nr.3, gültig ab 1.1.88
ISSN 0932-0385

Grafische Gestaltung:
Bernd Weber, Kerstin Feist

Fotografie:
Archiv, K.Ohlenschläger

Produktion:
K.H.Hoffmann, B.Failer, S.Failer

Druck:
Ferling Druck GmbH

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 7,-, ÖS 56,-, SFr 7,-
Jahresabonnement: DM 70,-
Europ. Ausland : DM 90,- Luftpost : DM 120,-
In den Preisen sind die gesetzliche MWSt. und die Zustellgebühren enthalten.

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit seiner Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern dem Heim Verlag.
Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

COMPUTER DES JAHRES

